Development of Dynamic Bayesian Network for the Analysis of High-Dimensional Biomedical Data

Arinze Akutekwe

PhD

2017

Development of Dynamic Bayesian Network for the Analysis of High-Dimensional Biomedical Data

Arinze Akutekwe B.Eng, MSc

A thesis submitted in partial fulfilment of the requirements of the University of Northumbria at Newcastle for the degree of Doctor of Philosophy

Research undertaken in the Department of Computer Science, Faculty of Engineering and Environment

January, 2017

Abstract

Development of Dynamic Bayesian Network for the Analysis of High-Dimensional Biomedical Data

Inferring gene regulatory networks (GRNs) from time-course expression data is a major challenge in Bioinformatics. Advances in microarray technology have given rise to cheap and easy production of high-dimensional biological datasets, however, accurate analysis and prediction have been hampered by the curse of dimensionality problem whereby the number of features exponentially larger than the number of samples. Therefore, the need for the development of better statistical and predictive methods is continually on the increase.

The main aim of this thesis is to develop dynamic Bayesian network (DBN) methods for analysis and prediction temporal biomedical data. A two stage computational bio-network discovery approach is proposed. In the ovarian cancer case study, 39 out of 592 metabolomic features were selected by the Least Angle Shrinkage and Subset Operator (LASSO) with highest accuracy of 93% and 21 chemical compounds identified.

The proposed approach is further improved by the application of swarm optimisation methods for parameter optimization. The improved method was applied to colorectal cancer diagnosis with 1.8% improvement in total accuracy, which was achieved with much less feature subsets of clinical importance than thousands of features when compared to previous studies.

In order to address the modelling inefficiencies in inferring GRNs from time-course data, two nonlinear hybrid algorithms were proposed using support vector regression with DBN, and recurrent neural network with DBN. Experiments showed that the proposed method was better at predicting nonlinearities in GRNs than previous methods. Stratified analysis using Ovarian cancer time-course data further showed that the expression levels Prostrate differentiation factor and BTG family member 2 genes,

were significantly increased by the cisplatin and oxaliplatin platinum drugs; while expression levels of Polo-like kinase and Cyclin B1 genes, were both decreased by the platinum drugs. The methods and results obtained may be useful in the designing of drugs and vaccines.

Contents

List of Figures

List of Tables

List of Publications produced from PhD Thesis

A. Akutekwe, H. Seker, and S. Yang, ”In silico discovery of significant pathways in colorectal cancer metastasis using a two-stage optimisation approach,” *IET Systems Biology*, vol. 9, no. 6, pp. 294-302, 2015.

A. Akutekwe and H. Seker, ”Inference of nonlinear gene regulatory networks through optimized ensemble of support vector regression and dynamic Bayesian networks,” in 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Aug 2015, pp. 8177-8180

A. Akutekwe and H. Seker, Inferring the dynamics of gene regulatory networks via optimized recurrent neural network and dynamic bayesian network,” in 2015 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), Aug 2015, pp. 1-8.

A. Akutekwe and H. Seker, Two-stage computational bio-network discovery approach for metabolites: Ovarian cancer as a case study,” in IEEE-EMBS Inter- national Conference on Biomedical and Health Informatics (BHI), June 2014, pp. 97-100.

A. Akutekwe and H. Seker, A hybrid dynamic bayesian network approach for modelling

temporal associations of gene expressions for hypertension diagnosis,"
in 2014 36th Annual International Conference of the IEEE Engineering
in Medicine and Biology Society, Aug 2014, pp. 804-807.

A. Akutekwe and H. Seker, Particle swarm optimization-based bio-network discovery method for the diagnosis of colorectal cancer," in 2014 IEEE International Conference on
Bioin- formatics and Biomedicine (BIBM), Nov 2014, pp. 8-13.

A. Akutekwe, H. Seker, and S. Iliya, An optimized hybrid dynamic Bayesian network approach using differential evolution algorithm for the diagnosis of hepatocellular carcinoma," in 2014 IEEE 6th International Conference on Adaptive Science Technology
(ICAST), Oct 2014, pp. 1-6.

A. Akutekwe and H. Seker, Two-stage bioinformatics approach for the diagnosis of hepatocellular carcinoma and discovery of its bio-network," in International
Conference on Applied Informatics for Health and Life Sciences
(AIHLS), October 2014, pp. 91-94.

To God Almighty, the faithful one for His grace and favour throughout this study and to my parents Engr.(Sir) & Lady (Dr.) O.C Akutekwe, and siblings for their love and support.

List of Abbreviations

DBN	Dynamic Bayesian Network
LASSO	Least Absolute Shrinkage and Selection Operator
DREAM	Dialogue for Reverse Engineering Assessments and Methods
PSO	Particle Swarm Optimisation
DE	Differential Evolution
HMM	Hidden Markov Models
KFM	Kalman Filter Models
RF	Random Forest
SVM	Support Vector Machine
RFE	Recursive Feature Elimination
RNN-DBN	Recurrent Neural Network Dynamic Bayesian Network
GRNs	Gene Regulatory Networks
CLPSO	Comprehensive Learning Particle Swarm Optimization
AUPR	Area Under the Precision Recall
AUROC	Area Under the Receiver Operating Characteristics
SVR-DBN	Support Vector Regression Dynamic Bayesian Network
SVR	Support Vector Regression
DAG	Directed Acyclic Graph
MTS	Multivariate Time Series
VAR	Vector Autoregression
LOOCV	Leave-One-Out Cross Validation

TFs	Transcription Factors
DNA	Deoxyribonucleic acid
mRNA	Messenger Ribonucleic acid
RBF	Radial Basis Function

List of Symbols

w	The weight vector of SVM
b	Bias of the SVM
G	A Network Graph
ε	Vector of white noise
X	Random Variable/population
$\mathbf{X}$	Stochastic Process
M	Transition Matrix
$\sum$	Covariance Matrix

Acknowledgements

I would like to sincerely thank my first supervisor Dr Huseyin Seker who gave me immense support, excellent guidance and great encouragement throughout this PhD. Words cannot describe my gratitude. I will also like to thank my second supervisor Prof. Shengxiang Yang for his assistance and guidance.

Declaration

I declare that the work contained in this thesis has not been submitted for any other award and that it is all my own work. I also confirm that this work fully acknowledges opinions, ideas and contributions from the work of others. This thesis produced eight research publications which have been submitted before in IET Systems Biology Journal and various IEEE conferences as listed in the list of publications section.

Any ethical clearance for the research presented in this thesis has been approved. Approval has been sought and granted by the Faculty Ethics Committee on 25th April 2013.

I declare that the Word Count of this Thesis is 25896 words

Name: Arinze Akutekwe

Signature: A.A

Date:18th January 2017

Chapter 1

Introduction

1.1 Motivation

Advances over the past few decades has resulted in increase in availability of both static and temporal high-dimensional biomedical data. The sequencing of the human genome has also increased research interest in developing novel methods for biomedical data analysis since its sequencing more than a decade ago [7], [8]. The discovery of more than three billion base pairs in the human genome project was undoubtedly remarkable. The success of the project heavily relied on computational methods which makes the field of computational biology very essential for future biological discoveries.

The completion of the human genome project gave rise to a new era known as post-genome era and advances in technology has resulted in the generation of vast amounts of data. A considerable amount of effort is required to discover knowledge from these data that may be of clinical benefit. Manual analysis in wet laboratories can no longer cope with the complexity of problems associated with the huge amount data and computational and statistical data mining approaches are inevitable.

The inevitability of computational methods for knowledge discovery in the vast amount of biomedical data since the post-genome era gave rise to the field of study known today as Bioinformatics. Bioinformatics is an interdisciplinary research field concerned mainly with the development of methods and software for mining and understanding of biological data. It integrates mathematics, statistics, computer science and biotechnology that aid in molecular sequence analysis, structural analysis and functional analysis of biomedical data.

These methods can be broadly divided into two. Methods for static analysis and methods for temporal or time-course analysis. Static data have been analysed using computational methods such as Boolean Networks and Artificial Neural Networks [9] but there is a lack of efficient methods for the analysis and visualization of high-dimensional temporal biomedical data such as Protein-Protein Interaction (PPI) networks [10]. Modelling and making predictions over time in non-linear complex systems involve the use of temporal (dynamic) data and not static data [11]. Various methods have been used to model the stochastic nature of such temporal data such as Hidden Markov Models (HMM), Kalman Filter Models (KFM) and DBN [12]. HMM and KFM are however not suitable for representing temporal data with two or more state variables, and large qualitative discrete variables, respectively [13].

A major problem of HMM and the motivation for choosing DBN is computational complexity. For example in an HMM where each state generates an observation with $X_t \in 1,...,K$ representing the hidden state at time t and y_t representing the observation, inference takes $O(TK^2)$ time for T sequence length and K number of states. For a factorial HMM with D chains each with K values, computing the probability $P(X_t|X_{t-1})$ needs $O(K^{2D})$ parameters to specify however, a DBN needs $O(DK^2)$ parameters to specify. For an HMM, the exact inference computational complexity is $O(TK^{2D})$ whereas a DBN is $O(TDK^{D+1})$ which is exponentially less than that of the HMM [13].

DBN represents temporal probability distribution over semi-infinite collections of random variables have widely gained recent attention [14]. However, optimisation of inference algorithms in DBN to handle high-dimensional correlated data using machine learning techniques have not been fully explored. Another motivation is that DBN-based methods for inferring the structure of the relationships among high quality selected biomarkers using high-dimensional datasets have not been explored as current methods focused on classification and feature selection performances without considering possible temporal relationships across selected biomarkers. Also, existing DBN methods assume that the relationships between features in time-course gene expression data are linear. This strong assumption may not always be true as biological networks are inherently nonlinear.

Time-course data refers to data from evolution of measurements over time from biological experiments. This differs from time-series data which refers to that from a sequence ordered in time. For instance time-course data could be from experimental measurements taken at different times such as 1hr, 3hrs, 9hrs and 24hrs as the experiment progresses while time series data is in an ordered pattern such as in speech recognition and signal processing. There is therefore need to develop a hybrid DBN that combines nonlinear predictive algorithms within a DBN. The parameters of the predictive algorithm could further be optimised using parameter optimisation algorithms for improved accuracy. This would lead to novel optimised hybrid DBN-based algorithms that would more accurately model and infer gene regulatory networks from time-course gene expression data. These developed algorithms could aid in the diagnosis of diseases such as cancer as the network of the disease metastasis could be more accurately inferred.

1.2 Contributions of the PhD Study

This thesis aims at developing and improving Dynamic Bayesian Network (DBN) for temporal prediction of biomedical data using optimisation and machine learning methods. DBN model was preferred as HMMs are more complex and are not suitable for representing temporal data with two or more variables and Kalman filters cannot be used to represent large qualitative discrete variables.

The contributions made are in line with the stated aim of the thesis.

- Development of a two-stage DBN-based bio-network discovery approach for analysing and inferring temporal associations of biomedical data.

  Accurate modelling of temporal relationships between biomarkers is important in disease prognosis and drug discovery. Existing methods focused only on classification performance of selected biomarkers but did not consider possible temporal relationships among feature subsets. At the first stage of the approach, feature selection is carried out using five different selection algorithms which are Random Forest Recursive Feature Elimination (RF-RFE), Least Absolute Shrinkage and Selection Operator (LASSO) and Linear, Polynomial and Radial Basis Function kernels of the Support Vector Machine Recursive Feature Elimination (SVMRFE) algorithm. Performance of the selected biomarkers were evaluated using machine learning criteria such as Sensitivity, Specificity, Accuracy, False Positive Rate and Matthews Correlation Coefficient. At the second stage, the temporal relationships of features with the best overall performance from the first stage are inferred using Dynamic Bayesian Network. The relevance of the feature interactions and inferred network model is verified from literature for each of the use cases studied. Two case studies that were developed and published to address this objective are: ovarian cancer [3] and hypertension [4]. This objective is also addressed in chapter 5.

- Development of optimised two-stage approach using swarm optimisation for parameter optimisation

  This study aimed to address the problem of inaccuracies in selection of key biomarkers of potential relevance to drug discovery by using optimization algorithms to fine-tune the parameters of feature selection algorithms for improved accuracy. Two key case studies were investigated:

  A) Diagnosis of Colorectal Cancer. In this study, significant pathways in colorectal cancer metastasis were discovered using the proposed two-stage DBN-based optimization approach. Previous modelling techniques from literature adopted less accurate filter methods of feature selection and methods of parameter optimization for improved accuracy were not proposed nor implemented. Furthermore, temporal relationships among potential biomarkers which might reveal significant pathways in the spread of the disease were not considered. This study aimed to address the aforementioned problems by adopting three kinds of more efficient SVMRFE feature selection algorithm. The algorithms were further optimized using particle swarm optimization and five differential optimization algorithms. The best performing features from the algorithm were modelled using DBN. The resulting inferred network, verified from published literature, showed that Alpha-2-HS-glycoprotein was highly associated with Fibrinogen alpha chain which had been shown to be a possible biomarker for colorectal cancer. This is addressed in chapter 6 and published in IET Systems Biology [15].

  B) Diagnosis of Hepatocellular Carcinoma (liver cancer)

  In this study, five Differential Evolution (DE) algorithms were used to optimize the parameters of three SVMRFE algorithms. To minimize the average error rate and increase accuracy, the algorithms: DE/rand/1, DE/best/1, DE/rand-to-best/1, DE/best/2 and DE/rand/2, were run for 30 independent times with each run done for 40,000 fitness evaluations. The Wilcoxon rank sum test was used

to evaluate algorithmic performance under the null hypothesis that the performance of the reference algorithm, DE/rand/1, is the same as the other DE algorithms using 95 percent confidence interval. The results obtained showed that features selected by DE/best/2-optimized SVMRFE radial kernel algorithm outperformed the others. The DBN inferred network analysis from the features showed that the SPINT2 gene may inhibit HGF activator which prevents the formation of active hepatocyte growth factor which are the chief functional cells of the liver. The methods and results of this study were published in IEEE [16].

- Development and implementation of optimized Recurrent Neural Network Dynamic Bayesian Network (RNN-DBN) algorithm for accurate inference of Gene Regulatory Networks (GRNs).

  The study aimed to address the problem of linearity assumed by most temporal inference methods and inefficiencies of parametric methods such as the S-systems model. After analytical reviews of different recurrent neural network models, the Elman Recurrent Neural Network was chosen due to its simple feedback connections which makes it powerful for nonlinear mapping. The parameters of the RNN were further optimized using particle swarm optimization (PSO) and comprehensive learning particle swarm optimization (CLPSO) algorithms. Results from the area under the precision recall (AUPR) curve using benchmark Drosophila Melanogaster dataset showed that the algorithm outperformed existing G1DBN algorithm which had been know to outperform three other algorithms: the LASSO, the ARACNE and the CLR algorithms. The developed algorithm was further used to model time-course human ovarian carcinoma cell dataset. Five key hub genes with four outgoing edges each were discovered. These are flap structure-specific endonuclease 1, kinesin family member 11, CDC6 cell division cycle 6 homolog (S. cerevisiae), histone 1, H2bd and TRAF family member-associated NFIB activator. This study is presented in chapter 7 and published in IEEE [5].

- Development and implementation of optimized Support Vector Regression Dynamic Bayesian Network (SVR-DBN)

  This study aims to further address the assumption of linearity by many gene regulatory network inference models proposed in literature. The study improves on DBN for modelling GRNs by using a more efficient non-linear support vector regression (SVR) algorithm. The SVR-DBN algorithm is further improved by the use of particle swarm optimization to fine-tune the parameters of the SVR. The robustness of the algorithm was tested on eight popular benchmark Dialogue for Reverse Engineering Assessments and Methods (DREAM) datasets and two real world datasets of Drosophila Melanogaster and Escherichia Coli. The results based on computed area under the precision recall curve (AUPR) and area under the receiver operating characteristics curve (AUROC) showed that the algorithm outperformed the existing G1DBN algorithm on all ten datasets. This study is presented in chapter 7 and published in IEEE [6].

The open-source R language for statistical computing was the programming language used for the whole of this PhD study [17]. Visualisation of inferred networks of biomedical data used was done with open-source visualisation software called Cytoscape [18].

1.3 Structure of the PhD Thesis

The remainder of the PhD thesis is structured as follows:

Chapter 2 discusses the background theory in computation methods used. It discusses machine learning methods such as the support vector machine (SVM), random forest and recurrent neural networks. It also discusses the support vector machine recursive feature elimination SVMRFE and shows how it integrates with optimization algorithms and how the linear, radial and polynomial kernels perform with varying sizes

of datasets. Bayesian networks and dynamic Bayesian networks are explained. The assumptions surrounding DBN modelling based on vector autoregressive models are explained. Furthermore, optimisation algorithms used in this thesis are also discussed.

Chapter 3 reviews literature in reverse engineering in Bioinformatics. Top-down and bottom-up methods are explained and how they are important to causal relationships. The generic framework of reverse engineering is reviewed and presented. Literature around modelling uncertainty using a Bayesian Network and the limitations of using a Bayesian Network are presented. Literature about temporal modelling is discussed. The limitations of other methods such as the Hidden Markov Model and Kalman Filter Models are explored and how dynamic Bayesian network is better than other methods. The chapter also explores some machine learning concepts in Bioinformatics and ends by explaining the gap in literature around inferring temporal relationships of high quality selected features and the problem with DBN representation that assume linear relationships among features.

Chapter 4 describes some of the datasets used in this study as contained within this thesis. It describes ovarian cancer metabolite dataset from [19] which contained a total of 592 metabolomic features from 72 observations. It discusses the hypertension gene expression dataset from [20] which contained a total of 22184 gene expression profiles and 159 observations. The chapter further describes colorectal cancer protein profiles datasets from [21] which had a total of 16331 spectral feature and 112 observations. Time-course ovarian cancer dataset from [22] which had 12625 genes is discussed and how 34 genes used were selected. Benchmark reverse engineering datasets which include both simulated and real datasets are also described.

Chapter 5 looks at the development of the two-stage approach for inferring the dynamics an temporal relationships in biomedical data. Feature selection methods used at the first stage include random forest recursive feature elimination, least absolute shrinkage

and selection operator (LASSO) and support vector machine recursive feature elimination. Cross-validation results are presented and discussed. Two DBN methods used at the second stage are compared. Two case studies of using ovarian cancer metabolites and hypertension gene expression dataset are discussed.

Chapter 6 explores ways of improving the developed two-stage approach. The two-stage approach relied on manual adjustment of parameter values such as the C and γ of support vector machine. Parameter optimization methods are discussed. In particular, two optimisation methods were adapted. These are the particle swarm optimisation (PSO) and the Differential Evolution (DE) algorithm. Results showed improvements over non-optimised approaches. The DBN was used to infer temporal relationships between selected variables.

Chapter 7 discusses the development of two DBN-based algorithms. The development was necessary as DBN methods previously adapted assumed linear relationship among features which was not necessarily a true representation of biological systems. Hybrid nonlinear approaches were therefore required. The recurrent neural network dynamic Bayesian network (RNN-DBN) was developed. This was efficient but computationally more intensive. To address that problem, the support vector regression dynamic Bayesian network (SVR-DBN) algorithm was then developed. The algorithms allowed for parameter optimisation which leads to more accurate inference.

Chapter 8 summarises and concludes the thesis. it discusses achievements and contribution to literature as well as some future works.

Chapter 2

Background Theory in Computation Methods

2.1 Introduction

The existence of things in the real world comprise of complex systems that evolve over time and respond to external perturbations. For biological systems, there has to be a way to capture these temporal evolution in order to infer progression of change in states of a system across time points. For instance the life cycle of an organism or the metastasis of a disease needs to be taken into account of. Machine learning techniques such as Support Vector Machines (SVMs) have long been used in bioinformatics for selecting features which may potentially be biomarkers for a disease. However, how these features may possibly relate to each other if each observation is taken as a time point has not been studied.

This is the idea behind the two-stage bio-network discovery approach introduced in this thesis and successfully applied and published in various case studies [3],[23], [4] using various datasets. Various datasets (described in chapter 4) of various dimensions were explored because they all have different nature and come from different tissues

and different genome platforms. This showed that the methods developed in this thesis would be able to fit into all these different kinds of datasets and can handle both small and high-dimensional datasets.

This chapter seeks to explore and define the background theory of the computational methods used in this thesis.

2.2 Dynamic Bayesian Network (DBN)

Static Bayesian networks are graphical models that allow for the representation of probabilistic dependencies between a given set of random variables in a concise way. The random variables $x = x_1, x_2, ..., x_p$ are represented as a directed acyclic graph (DAG). Real world entities are however made up of complex systems which change over time. Dynamic Bayesian Networks extends static Bayesian network in its ability to capture temporal informations which are useful in modelling feedback loops. As this feedback loops are common in biological pathways, DBNs modelling is expected to capture better representation of these pathways.

DBNs can be modelled from multivariate time series (MTS) based on vector auto-regression (VAR). MTS is used to model interactions among a group of time series variables. MTS are usually modelled as vector auto-regressive processes [24]. For a vector auto-regressive process VAR of order p denoted as VAR(p), variables observed at any time $t \geq p$ satisfy the equation:

$$X(t) = A_1 X(t-1) + ... + A_i X(t-i) + ... + A_p X(t-p) + B + \varepsilon(t) \qquad (2.1)$$

where

- $X(t) = (Xi(t)), i = 1, ..., k$ is the vector of k variables observed at time t;

- $A_i, i = 1, ..., p$ are matrices of coefficients of size $k \times k$;
- B is a vector of size k which represent the baseline measurement for each variable;
- $\varepsilon(t)$ is a white noise vector of size k with zero mean $(E(\varepsilon(t)) = 0)$ and time invariant positive definite matrix $(COV(\varepsilon(t)) = \sum)$

In the above MTS representation, VAR of order p assumes a linear relationship between the k variables observed at t time points and k variables observed at previous time points p. In the DBN, each variable is represented by several nodes across time points. A DBN is obtained when an interaction graph is unfolded in time which enables the accommodation of feedback loops. The acyclicity of the graph is required to ensure a Bayesian network representation is maintained by setting the arc variables in time. The arc is drawn between two successive time points in the resulting graph.

To enable a DBN to be represented as a VAR process, various assumptions are made [1].

Assumption 1: The stochastic process **X** is assumed to be of first-order Markov chain.

This ensures that variable at time t depends on the past only through variables that were observed at time $t-1$.

Assumption 2: The random variables $X(t) = (X_1(t), ..., X_i(t), ..., X_k(t))$ observed at time t, for all $t > 0$, are conditionally independent given the random variables $X(t-1)$ at the previous time $t-1$.

This implies that simultaneously observed variables at any time point are conditionally independent given their immediate past variables. Hence a variable X_i at time t is better explained by $X(t-1)$ than by any other variable X_j at the same time point. This is based on the assumption that time points are close enough.

Assumptions 1 and 2 allows for a DBN with graph G to be modelled which will have arc pointing from a variable observed at time $t-1$ to a variable observed at time t and no arcs between variables observed simultaneously. A constant time delay for all the interactions is assumed to restrict the number of parameters of the network. It may be possible to include simultaneous interactions or a longer time delay e.g. from t to $t-2$ however, this will increase the number of parameters exponentially. This is left as a future study.

Assumption 3: For any variable X_i, the temporal profile $X_i(1),...,X_i(n)$ cannot be written as a linear combination of other profiles $X_j(1),...,X_j(n)$, where $j \neq i$.

The uniqueness of the graph G is guaranteed by this assumption when the k variables are linearly independent. This means that the profiles cannot be linearly combined. Assumptions 1, 2 and 3 guarantees that the probability distribution of the stochastic process $\mathbf{X}$ can be represented as a DBN.

Theorem 1

The probability distribution of $\mathbf{X}$ can be represented as a dynamic Bayesian network with directed acyclic graph G whenever assumptions 1, 2 and 3 are satisfied. The arcs of G describe exactly the conditional dependencies between any pair of variables at successive time points given any past variables [1].

Assumption 4: The process is homogeneous over time; all arcs and their directions are time invariant.

This assumption allows for proper representation of a MTS using small number of parameters. For each variable at a given time point, large number of repeated measurements are needed for estimation. This kind of data is rarely available and in bioinformatics; most gene expression time series contain very few or no repeated measurements [25].

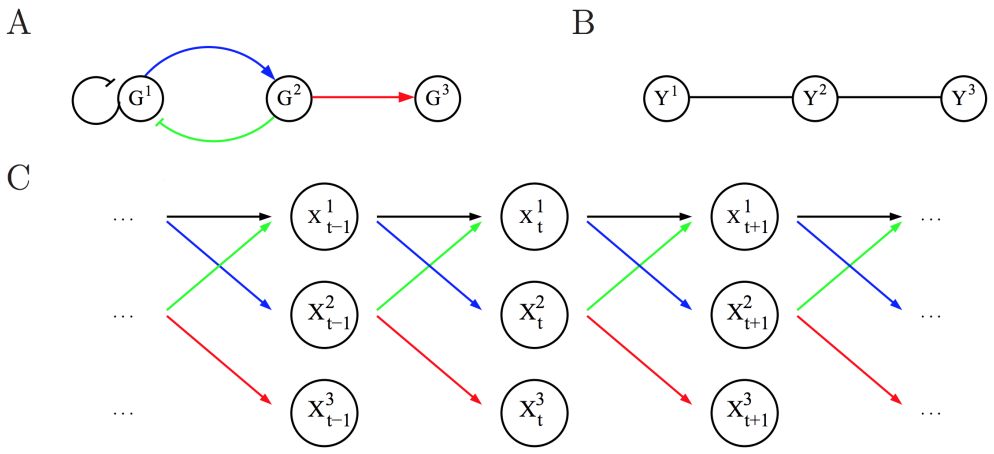

FIGURE 2.1: Dynamic Bayesian Network of a Regulation Motif [1]

Part A of Figure 2.1 shows a biological regulation motif with genes G^1, G^2 and G^3. Part B is the Gaussian variable representation Y^i of the expression level of G^i for all $i \geq 3$. Part C shows the dynamic network equivalent of the regulation motif A where the expression level of gene G^i at time t is represented by vertex X_t^i.

If a VAR process of order 1 is assumed, a DBN can be represented as

$$X_t = MX_{t-1} + B + \varepsilon(t) \tag{2.2}$$

where M is a $k \times k$ transition matrix which expresses dependence of X_t on X_{t-1}, $\varepsilon(t)$ is a vector white noise assumed to be multivariate normal with zero mean and covariance matrix $\sum$. B is a k column vector representing baseline measurement for each variable. The arc is set by defining the set of non-zero coefficients in matrix M such that if the element a_{ij},$i \neq j$ differs from zero, an arc will be included in the network from $X_i(t-1)$ to $X_j(t)$. Off diagonal elements in the covariance matrix $\sum$ can be set to zero if the error term for each variable X_i is independent of other variables.

$$M = \begin{bmatrix} a_{11} & a_{12} & 0 \\ a_{21} & 0 & 0 \\ 0 & a_{32} & 0 \end{bmatrix} \tag{2.3}$$

Learning a DBN defining a VAR process is akin to identifying the non-zero coefficients of the auto-regressive matrix M. From theorem 1, a VAR process of order 1 denoted as VAR(1) whose error covariance matrix $\sum$ is diagonal can be represented as a DBN whose non-zero elements of M identify the arcs to be included in the network.

2.3 Support Vector Machine

First introduced by Cortes and Vapnik in 1995 [26] the SVM is a supervised learning method for classification and regression. For classification tasks the goal of the SVM is to construct a hyperplane or a set of hyperplanes in a high-dimensional space to be used for classification of a linearly separable dataset. The hyperplane with largest distance to the training data is said to achieve good separation. The SVM can also be used for non-linearly separable datasets which are common with high-dimensional datasets.

For a two class classification problem with training vectors $x_i = i, ...l$ in a Euclidean space, an indicator vector y can be defined such that

$$y_i = \begin{cases} 1 \, if \, x_i \text{ in class 1} \\ -1 \, if \, x_i \text{ in class 2} \end{cases} \tag{2.4}$$

The aim of the SVM therefore is to find a separating hyperplane which separates all the data as shown in Figure 2.2.

The hyperplane $w^T x + b = 0$ obeys the inequality constraints

$$w^T x_i + b > 0 \text{ if } y_i = 1 \tag{2.5}$$

$$w^T x_i + b > 0 \text{ if } y_i = 1 \tag{2.6}$$

where the weight w and the bias b of the separating hyperplane need to be optimally decided for good test set prediction. The most general hyperplane is the one that maximizes the width of the hyperplane which is the distance of two parallel lines. The distance between the lines $w^T x_i + b = 1$ and $w^T x_i + b = -1$ is maximised by $2/\|w\|$ = $2/\sqrt{w^T w}$ called the maximal margin. The SVM therefore aims to find w and b such that the value of $2/\|w\|$ is maximised.

Maximising a function however is equivalent to minimising its reciprocal and the increasing function (the square root) is therefore removed [27] which results in:

$$\min_{w,b} \left\{ \frac{1}{2} w^T w \right.$$

subject to the constraints of Equation 2.5 and Equation 2.6. Combining the two inequalities and multiplying y_i on both sides of Equation 2.5 and Equation 2.6 gives rise to the quadratic programming problem

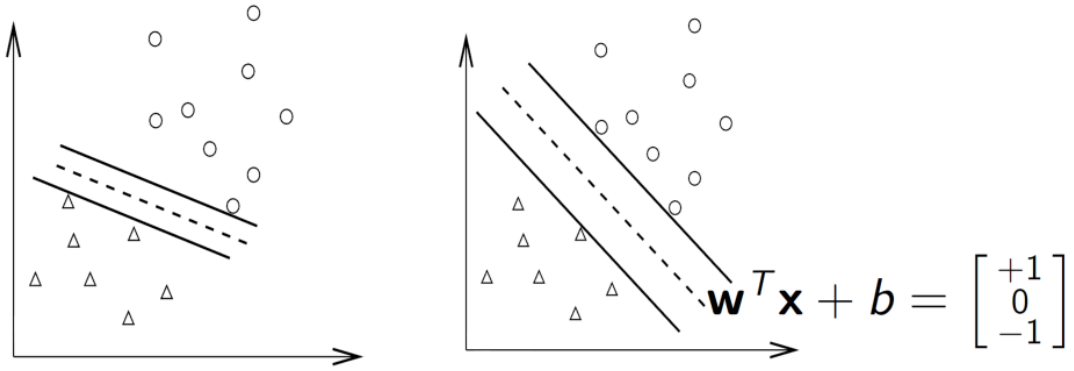

FIGURE 2.2: The SVM hyper-plane showing linearly separable data points.

$$\min_{w,b} \left\{ \frac{1}{2} w^T w \right.$$

subject to $y_i(w^T x_i + b) \geq 1; i = 1, ..., l$ where l is the length of the training vectors. For nonlinearly separable dataset, a nonlinear curve can be used to fit the data where training errors are allowed in order not to have an infeasible optimisation problem. In order to avoid many training errors, a penalty function $C \sum_{i=1}^{l} \zeta_i$ is introduced where C is the cost of constraint; a large value of C results in small penalty, and ζ is a positive slack variable. This results in the standard SVM function

$$\min_{w,b,\zeta} \left\{ \frac{1}{2} w^T w + C \sum_{i=1}^{l} \zeta_i \right.$$

subject to $y_i(w^T \phi(x_i) + b) \geq 1 - \zeta_i$, $\zeta_i \geq 0, i = 1, ..., l$ where ϕ is a nonlinear function. Solving the inner product of two vectors in high dimension presents a difficult problem but can be handled effectively by kernels. When mapped to a high-dimensional feature space, kernels return the value of the dot product between the images of two arguments originally in an input space. x is mapped from an input space to a transformed feature space $\phi(x)$ where data is separable. This results in a dual classifier in a transformed feature space where $\phi(x)$ only occurs in pairs $\phi(x_j)^T \phi(x_i)$.

Different kernels K result in different kinds of SVM. Three popular kernels are used in this thesis:

The linear kernel:

$$K(x_i, x_j) = \phi(x_i).\phi(x_j) \tag{2.7}$$

The Polynomial kernel:

$$K(x_i, x_j) = (x_i.x_j + 1)^d \tag{2.8}$$

where d is the dimension of the polynomial.

The Radial Basis Function kernel:

$$K(x_i, x_j) = \exp^{-\gamma \left\| x_i - x_j \right\|^2} \tag{2.9}$$

The SVM recursive feature elimination (SVM-RFE) uses the magnitude of the weight of the SVM classifier to produce feature ranking by backward elimination [28] and hence more accurate feature selection is performed because the algorithm interacts with the classifier. The recursive feature elimination involves the following procedure:

1. Training the classifier.
2. Calculating the ranking criterion.
3. Removing the features with the smallest ranking criteria.

For regression tasks, the support vector machine for regression or the Support Vector Regression (SVR) has the goal of finding a function $f(x)$ that has the most ε deviation from the actually obtained targets y_i, where ε is the tolerance error. For the regression model, only errors less than ε are acceptable and the SVR can be formulated with different constraints as:

$$\min_{w,b,\zeta,\hat{\zeta}_i} \left\{ \frac{1}{2} w^T w + C \sum_{i=1}^{l} (\zeta_i + \hat{\zeta}_i) \right.$$

subject to:

$$y_i - w^T \phi(x_i) - b \leq \varepsilon + \zeta_i \tag{2.10}$$

$$(w^T \phi(x_i) + b) - y_i \leq \varepsilon + \hat{\zeta}_i \tag{2.11}$$

$$\zeta_i, \hat{\zeta}_i \geq 0 \tag{2.12}$$

where $\zeta_i, \hat{\zeta}_i$ are positive slack variable used to account for training errors. This approach is called the ε-Support Vector regression (or ε-SV regression) [29] and was adapted in chapter 7 of this thesis because it is the most common approach and performs well in high-dimensional dataset.

2.4 Random Forest

The random forest algorithm was developed by Leo Breiman [30] and Adele Cutler [31] for tasks of classification and regression. It is an ensemble learning learning algorithm that works by generating many decision trees using Breiman's idea of bagging in tandem with random selection of features introduced by Kam [32] and Ho [33]. Many advances in the area of random forest has also come from Microsoft Researchers Criminsi, shotton and Konukoglu [34] which extend the earlier work done by Breiman.

The algorithm works by applying bagging to ensemble of decision trees. For a training set $X = x_1, ..., x_n$ with responses $Y = y_1, ..., y_n$, a bootstrap sample of the training set is repeatedly selected by bagging and decision trees are fit to them. For samples b = 1 through B, sample with replacement n training examples from X,Y which will be called X_b, Y_b. The random forest function can then be written as:

$$\hat{f} = \sum_{b=1}^{B} \frac{1}{B} \hat{f}_b(x') \tag{2.13}$$

where B is a free parameter that represents trees to be trained on different subsets of the data and $\hat{f}_b$ is the b'th tree. A regression or decision tree f_b is trained on X_b, Y_b.

Predictions for unseen sample x' can be made after training by averaging predictions from all the individual regression trees on x'. In the case of decision trees for classification tasks, the majority vote is taken. Example if in four decision trees within the forest, two belonged to class y_1, one to class y_2 and one to class y_3, class y_1 is chosen since it score the majority vote. Increasing the number of trees decreases the variance without increasing the bias. This makes the training and test error to remain constant after fitting some number of trees. Cross-validation can be used to find the optimum number of trees B. The authors in [35] also suggested that finding the optimal number of trees can also be achieved by observing out-of-bag error which is the mean prediction error on each sample x_i using trees that did not have x_i in their bootstrap sample.

2.5 Recurrent Neural Network (RNN)

The class of Artificial Neural Network where forward and feedback (recurrent) connection between neurons exist is known as Recurrent Neural Neural Network. Recurrent Neural Network allow feedback connections unlike Multilayer Perceptron that allow only feedforward connections between neurons across different layers. This makes them powerful and have been used to infer gene regulatory network from gene expression data [36]. Temporal dynamics of a system can be represented by an a RNN and can be used to model and predict nonlinear and noisy time-course gene expression than linear predictors. There are four main Recurrent Neural Network architectures commonly used [37].

2.5.1 Input-Output Recurrent Model

In this architecture, an input is applied to k units of tapped-delay-line memory. A single output is fed back to the inputs through another k units of tapped-delay-line

memory. The inputs layer of the multilayer perceptron is then fed by the contents of these tapped-delay-line memories. The result is a nonlinear autoregressive with exogenous inputs (NARX) model whose dynamic behaviour can be represented by

$$Y_{n+1} = \theta(Y_n, Y_{n+1}, ..., Y_{n-q+1}; U_n, U_{n+1}, ..., U_{n-q+1}) \tag{2.14}$$

where θ is a nonlinear function of its arguments, U_n is the input to the model and $U_n, U_{n+1}, ..., U_{n-q+1}$ represent the present and past values of the exogenous inputs. $Y_n, Y_{n+1}, ..., Y_{n-q+1}$ represents the delayed values of the output on which the model output Y_{n+1} is regressed.

2.5.2 The Recurrent Multilayer Perceptron (RMLP)

The Recurrent Mutlilayer Perceptron architecture has one or more hidden layers with each layer having a feedback and can be more effective than using only single hidden layer. If vector $X_{I,n}$ denotes the output of the first hidden layer, $X_{II,n}$ denotes the output of the second hidden layer and vector $X_{o,n}$ denotes the overall output of the output layer, then the system of coupled equations given as

$$\begin{aligned} X_{I,n+1} &= \theta_I(X_{I,n}, Un) \\ X_{II,n+1} &= \theta_{II}(X_{II,n}, X_{I,n+1}) \\ &\vdots \\ X_{o,n+1} &= \theta_o(X_{o,n}, X_{K,n+1}) \end{aligned} \tag{2.15}$$

represents the general dynamic behaviour of the RMLP to U_n input vector of K hidden layers. $\theta_I(.,.)$, $\theta_{II}(.,.)$ and $\theta_o(.,.)$ represent the activation functions of the first hidden layer, the second hidden layer and the output layer of the network respectively.

2.5.3 Second-Order Network

Second-order network comprise of second-order neuron k that uses a single weight w_{kji} to connect it to input nodes i and j. They are the result of multiplying first-order networks where order refers to the way in which an induced local field of a neuron is defined. Second-order network accepts time-ordered sequence of inputs and evolve with dynamics defined by the pair of equations:

$$v_{k,n} = b_k + \sum_i \sum_j w_{kij} X_{i,n} u_{j,n} \tag{2.16}$$

$$X_{k,n+1} = \phi(v_{k,n}) \frac{1}{1 + exp(-v_{k,n})} \tag{2.17}$$

where $u_{j,n}$ is the input applied to source node j, w_{kij} is a weight of second-order neuron k, $v_{k,n}$ is the induced local field of the hidden neuron k, b_k is the associated bias and $x_{k,n}$ is the output of neuron k.

2.5.4 State-Space Model and the Elman Network

In the state-space model, the state of the network is defined by the hidden neurons. The output of the hidden layers is fed back into the input through a bank of unit-time delays which determine the order of the model. If vector u_n denotes the input vector and vector x_n denotes the output of the hidden layer at time n, then the dynamic behaviour of the model may be described by the pair of equations:

$$x_{n+1} = \alpha(x_n, u_n) \tag{2.18}$$

$$y_n = \beta x_n \tag{2.19}$$

where $\alpha(.,.)$ is a nonlinear function characterising the hidden layer and β is the synaptic weight matrix of the output layer. In the Elman network [38] also known as the simple recurrent network (SRN) which is derived from the state-space model, the output layer maybe nonlinear and the bank of unit-time delays at the output is omitted. The Elman network has an error derivative delayed by one time step back in the past and can be used to model time-course biomedical data more intuitively. The Elman network was adapted in chapter 7 of this thesis due to its state-space characteristic and simplicity. It best fits within the DBN framework due to its Markovian properties of modelling nonlinear relationships delayed by one time step in the past. As explained in chapter 7, the algorithm developed by combining it within DBN (RNN-DBN) was efficient at inferring gene regulatory network from time-course gene expression data.

2.6 Parameter Optimisation Algorithms

Optimisation algorithms generally try the find the minimum values of various mathematical functions. Parametric optimisation involves a search in the space spanned by parameter variables of a defined objective function with the aim of minimising or maximising the function. They are used to fine-tune parameter values of various classification or regression algorithms in scientific and engineering computations. They include evolutionary computation methods which are biologically inspired computational techniques based on theories of natural selection, reproduction and survival of the fittest. Some well-developed evolutionary computation methods include Genetic Algorithm [39], Genetic Programming [40], Evolution Strategies [41] and Differential Evolution (DE) [42]. Other optimisation algorithms are inspired by theories of collective intelligence such as Particle Swarm Optimisation (PSO) [43] where individuals

in a group (swarm) rely on each other's individual knowledge of a treasure's whereabouts in order to find that treasure. The two optimisation algorithms adapted in this thesis are Differential Evolution (DE) and Particle Swarm Optimization (PSO). They were used because at the time of this PhD study, PSO and DE were not much explored in the DBN domain for biomarker discovery and were quite good in high-dimensional data applications of this research.

2.6.1 Differential Evolution (DE) Algorithm

DE is a population-based meta-heuristic algorithm developed by Storm and Price [42] in which an objective function is optimised by iteratively searching through and improving candidate solutions. In DE, mutation step is applied before crossover by generating a trial vector and is not sampled earlier from a known distribution. The robustness of the algorithm has led to its application various industries such as mechanical engineering [44] and pattern recognition algorithms [45]. The algorithm aims at evolving a population X of D-dimensional parameter vectors known as individuals such that $X_{i,K} = X^1_{i,K}, ..., X^D_{i,K}$ for $i = 1, ...D$ [46]. The individuals are randomised within the search space such that the entire search space is covered as much as possible by the initial population. The population is constrained within the minimum bound $X_{min} = X^1_{min}, ..., X^D_{min}$ and maximum bound $X_{max} = X^1_{max}, ..., X^D_{max}$ such that the initial value of the jth parameter in the ith individual at generation $K = 0$ is given by:

$$X^j_{i,0} = X^j_{min} + rand(0,1) \times (X^j_{max} - X^j_{min}) j = 1, 2, ..., D \tag{2.20}$$

where rand(0,1) is a uniformly distributed random variable between 0 and 1. Mutation takes place after initialisation to produce a mutant vector $V_{i,K}$ with respect to each individual target vector $X_{i,K}$ at generation K. The mutant vector $V_{i,K} = v^1_{i,K}, v^2_{i,K}, ..., v^D_{i,K}$ can be generated by various variants of the DE algorithm [47]. Five main variants are:

DE/rand/1

$$V_{i,K} = X_{r_1^i,K} + F \times (X_{r_2^i,K} - X_{r_3^i,K}) \tag{2.21}$$

DE/best/1

$$V_{i,K} = X_{best,K} + F \times (X_{r_1^i,K} - X_{r_2^i,K}) \tag{2.22}$$

DE/rand-to-best/1

$$V_{i,K} = X_{r_1^i,K} + F \times (X_{best,K} - X_{r_3^i,K}) + F \times (X_{r_1^i,K} - X_{r_2^i,K}) \tag{2.23}$$

DE/best/2

$$V_{i,K} = X_{best,K} + F \times (X_{r_1^i,K} - X_{r_2^i,K}) + F \times (X_{r_3^i,K} - X_{r_4^i,K}) \tag{2.24}$$

DE/rand/2

$$V_{i,K} = X_{r_1^i,K} + F \times (X_{r_2^i,K} - X_{r_3^i,K}) + F \times (X_{r_4^i,K} - X_{r_5^i,K}) \tag{2.25}$$

where F is a scaling factor, the indices $r_1^i, r_2^i, r_3^i, r_4^i and r_5^i$ are mutually exclusive integers in the range 1 to D.

2.7 Particle Swarm Optimisation (PSO) Algorithm

The PSO is a stochastic population-based optimisation method inspired by the social behaviour of fish schooling and bird flocking which was developed by Kennedy and Eberhart [43] with a simple concept for each individual to emulate the successes of neighbouring individuals as well as its own success. This gives rise to discovery of optimal regions of a high-dimensional search space [46]. In the conventional paradigm

of evolutionary computation, a swarm is similar to a population, and a particle is similar to an individual. Since its inception, PSO has attracted a lot of research interests from DNA sequencing [48] to optimising path-following footsteps for humanoid robots [49].

The PSO adjusts its particles in searching the space of an objective function. A particle's position is changed by the addition of a velocity $v_i(t)$ to its current position such that

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{2.26}$$

where $x_i(t)$ represents the position of particle i in the search space at discrete t time steps. The optimisation process is driven by the change of the velocity and represents the experiential knowledge or cognitive process of each particle. A point in the D-dimensional space is represented by each ith particle such that $x_i = (x_{i1}, x_{i2}, ..., x_{iD})$. Different variants of the PSO exist such as the inertia weight PSO [50] and the comprehensive learning PSO (CLPSO) [51].

For velocity v of each particle i, the inertia weight PSO's velocity updating equation can be written as:

$$v_{id} = w \times v_{id} + c_1 \times rand() \times (p_{id} - x_{id}) + c_2 \times Rand() \times (p_{gd} - x_{id}) \tag{2.27}$$

$$x_{id} = x_{id} + v_{id} \tag{2.28}$$

where w is the inertia weight, the best known location or experience point is recorded as $P_i = p_{i1}, p_{i2}, ..., piD$ and c_1 and c_2 are positive acceleration constants. g represents the index of the best particle among all particles while $Rand()$ and $rand()$ are two

random functions in the range of 0 to 1. In the CLPSO, a particle's velocity is updated by all other particles' historical best information.

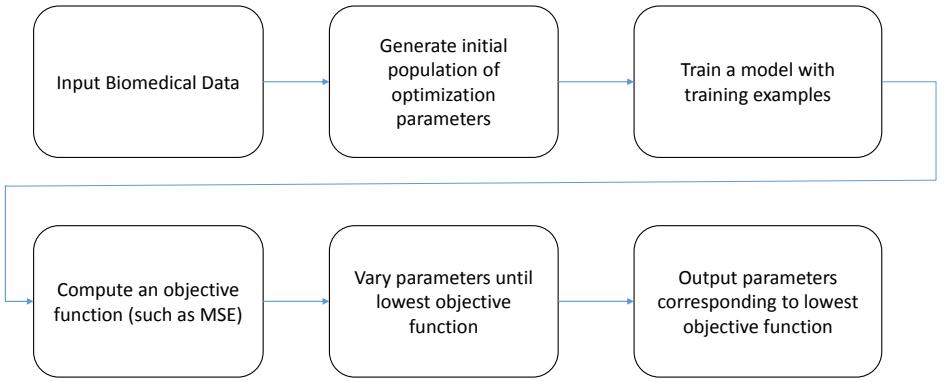

FIGURE 2.3: Generic Computational Optimization Model adapted in this study

Figure 2.3 shows the block diagram of generic computational optimisation model adapted in this study. It takes biomedical data as input, generates initial population of optimisation parameters. For the SVM for instance, this would be the C and γ parameters. The model is trained using the parameters while an objective function such as mean squared error (MSE) is being computed. The process continues with various optimisation parameters being varied within the optimization algorithm until lowest objective function is reached. The output is a vector of optimised parameters corresponding to the lowest objective function. More specific block diagrams of computational models adapted in this study are shown in their respective chapters.

2.8 Conclusion

This chapter introduced and explored the theory behind the computational methods used in this thesis. It describes support vector machine, Random Forest and the Recurrent Neural Network (RNN) algorithms. The random forest can be used for both classification and regression tasks and was used for the task of classification and feature selection using recursive feature elimination. The RNN was introduced as a method of modelling the temporal dynamics of a system and four architectures discussed which are the Input-Output Recurrent Model, State-Space Model and the Elman Network, the Recurrent Multilayer Perceptron and the Second-Order Network. The State-Space Model and the Elman Network was chosen and used to develop a novel Recurrent Neural Network Dynamic Bayesian Network (RNN-DBN) algorithm for the inference on gene regulatory network from time-course gene expression data. The choice was due to its simplicity, state-space and Markovian properties of modelling nonlinear relationships delayed by one time step back in the past.

Various parameters of algorithms such as Support Vector Machine and Recurrent Neural Network need to optimised and fine-tuned for best performance. Parameter optimisation algorithms were introduced which include Particle Swarm Optimisation and Differential Evolution algorithms. These two algorithms were adapted because at the time of this study, they have not been much explored within the DBN domain and were quite good in high-dimensional dataset case studies throughout this research.

Dynamic Bayesian Network (DBN) is described and the DBN describing a vector autoregressive (VAR) process is explained along with the theorem and necessary assumptions. Support Vector Machine is described along the three kernels used in this thesis: the linear, the polynomial and the radial basis function kernels. The SVM was both used for the tasks of classification and feature selection using the recursive feature elimination (RFE) algorithm by backward selection. The SVM was further used for the task of regression in chapter 7 (using nonlinear radial basis function kernel

within DBN) to develop novel Support Vector Regression Dynamic Bayesian Network (SVR-DBN) inference algorithm for reverse engineering of gene regulatory network from time-course gene expression data.

Understanding the state-of-the-art in reverse engineering of gene expression network from time-course gene expression data is therefore important in appreciating how DBN works and how it is applied for this task. This is discussed in chapter 3.

Chapter 3

Literature Review

3.1 Reverse Engineering in Bioinformatics

In protein synthesis, transcriptional regulation plays an important role in the response of the proteins to various internal stimulus like development processes and external stimulus from the environment [52]. Transcriptional regulations in the cell are responsible for partial and temporal levels of mRNA and the abundance of protein [53]. Regulatory activities between regulators such as transcription factor (TF) make up the transcriptional regulatory network. The expression of the genes is determined by the regulatory relationship's activation and repression [54].

Microarrays are technologies used to measure the amount of production of mRNA during transcription by hybridisation of an mRNA molecule to the source DNA template. The matching of both known and unknown DNA samples in an orderly arrangement constitutes an array and this array samples make up Microarray data. The expression level of a gene is the amount of mRNA that is bound to each site on the array. The microarray chip can identify spots with more intensity for a diseased tissue gene if in the diseased condition, that gene is over expressed. [55] [56] .

Identifying and understanding transcriptional regulatory networks is of great importance in discovering potential drug targets [57], [58]. In understanding these networks, various reconstruction methods have been proposed which generally fall into top-down and bottom-up methods. Top-down methods are those that identify the global regulatory interactions in parallel and systematic way by first acquiring many regulatory interactions and then through additional experiments validating them. As an example, identifying the interactions of protein-DNA is made possible by ChIP-Seq technology. It combines chromatin immunoprecipitation (ChIP) with massively parallel DNA sequencing for the identification of DNA-associated protein binding sites [59] [60], [61]. Bottom-up methods are gene knock-out experiments in which detailed regulations between transcription factors and their individual targets are are first identified, and then all the regulations are summarised to form a regulatory network. After knocking out some genes, the genetic relationship can then be detected as a result [62], [63].

Reverse engineering is the term used to describe the inference or reconstruction gene regulatory networks from gene expression data produced from microarray experiments [64]. Reverse engineering of gene regulatory networks have recently become very popular in computational biology and bioinformatics and numerous computational methods have been studied. [65] reviewed how complex networks are being constructed from simple modular components. It also reviewed how stochastic and deterministic modelling of the modules can provide more accurate *in silico* representation of gene regulatory networks. [66] reviewed how Bayesian statistical methods can be used to infer the network structure and estimate model parameters based on experimental data. [67] and [68] developed *in silico* reverse engineering benchmark datasets through community efforts of the Dialogue for Reverse Engineering Assessments and Methods (DREAM) project. These datasets known as the DREAM datasets are used to test new algorithms developed for reverse engineering of gene regulatory networks from time-course gene expression data.

[69] used the DREAM datasets to perform comprehensive assessment of various inference methods on Staphylococcus aureus, Escherichia coli, *in silico* microarray data and Saccharomyces cerevisiae. The authors discovered that no single inference method performed optimally across all datasets. [70] developed a new algorithm for inferring gene regulatory networks known as BAyesian Clustering Over Networks (BACON). The algorithm uses integrated, probabilistic clustering to lessen the problems of under-determination and correlated variables within a fully Bayesian framework. The authors of [71] suggest that morphogenesis may be reverse engineered to uncover its interacting mechanical pathway and molecular circuitry.

[72] developed a new search heuristic known as Divided Neighbourhood Exploration Search to be used with inference algorithms such as Bayesian Networks for improving inference in reverse engineering. The algorithm systematically moves through the search space to find topologies representative of gene regulatory networks that are more likely to explain microarray data. [73] used the DREAM datasets to identify a better prognostic model for prediction of survival in patients with metastatic castration-resistant prostate cancer. The computational methods for inference of these gene expression data are motivated by the availability of genome-wide profiling data. Gene perturbations or expression profiles of time series show dynamics of genes and imply the possibilities of causal relationships [74], [75].

3.1.1 The Generic Framework for Reverse Engineering in Bioinformatics

Since the advent of microarray technologies, there has been great amount of opportunities to elicit information and insights from genome-wide gene expression data by measuring and monitoring their expression patterns [55]. Regulatory knowledge can also be derived from experimental data using various strategies developed for inferring regulatory architectures from their corresponding gene expression profiles [68], [69].

The general framework for reverse engineering is illustrated in Figure 3.1A. The aim is to identify and infer relationships between transcription factors and target genes from their expression profiles. Regulators in the network are represented by the nodes and regulatory interactions are represented by the edges. From part A of the Figure 3.1, the network structures and parameters are reversely engineered from gene expression data which may be perturbation experiments of gene knockouts or time series processes from gene expression data. Models and reverse engineering algorithms are developed to measure these structure and parameters such as causality and strength of the relationships. For example, in the inference algorithm, the strength could be measured as p-values where lower values indicate greater strength of edges [1].

For genes G_1 and G_2, there are four levels of clarity for which four different questions about regulatory parameters need to be answered as shwon in Figure 3.1B. Level I seeks to determine whether there are regulatory interactions between G_1 and G_2 from the regulatory matrix data. When a causal influence from G_1 to G_2 has been identified at level I, level II determines the edge direction by determining which gene is the

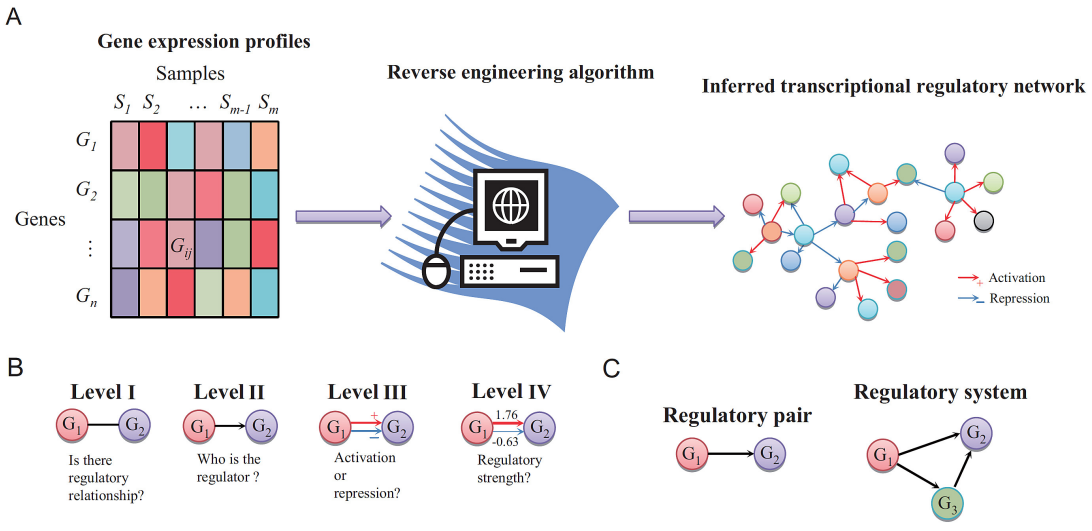

FIGURE 3.1: Generic Framework of Reverse Engineering of Regulatory Networks. Part (A) represents how gene expression profiles data are fed into a reverse engineering algorithm and the output is an inferred transcriptional regulatory network showing activation and repression. At (B), the algorithm addresses the four levels of clarity in the reverse engineering process. (C) shows regulatory pair and the regulatory system that consists of both complicated regulations such as regulation from G_1 to G_2 conditioned upon G_3 that has to be systematically modelled. [2]

regulator. Concentrations of target genes may be increased or decreased in certain conditions when transcription factors activate or repress the target gene. Level III determines whether the information is about activation or repression. For example, if M is the regulatory matrix, G_1 activates G_2 if $M_{12} > 0$ or $M_{21} = 0$ and G_1 represses G_2 if $M_{12} < 0$ or $M_{21} = 0$. Level IV determines the regulatory strength between G_1 and G_2. For example $M_{12} = 1.76$ or $M_{21} = -0.63$ shows the regulatory strength of the relationship between the genes. In part C of Figure 3.1, strength of the regulation can be determined as strong or weak by assessing all the real numbers of the regulatory strength. In the illustration, G_2 is regulated by G_1 and shows direct causality while the right shows an indirect causality is shown from G_3. When matrix M is very large, there obviously needs to be systematic and computationally efficient way of modelling these relationships [2].

One major difficulty associated with reverse engineering in bioinformatics is the curse of dimensionality problem where $p \gg n$ i.e. number of genes (predictors) p in thousands have been experimented with very few samples n [76]. Also the likelihood of false positives is very high since the regulatory networks are usually sparse [77], [78], [79], [80]. The dynamics of gene regulation as a result of its real environment also contribute to the difficulty in inference. For instance the downstream target genes can be affected by upstream regulation of a gene that encodes a transcription factor [81], [82]. There is also a lot of mismatch and inconsistency between protein and mRNA which affects the correctness of the regulation inference [83]. All these challenges pose significant difficulty in using expression data to reconstruct regulatory relationships.

In order to address these challenges in reverse engineering of gene regulatory networks and its evaluation, *in silico* methods have been developed to discover substantial regulations and traditional methods have been used to validate them [84], [85], [86], [87]. One of such is The Dialogue for Reverse Engineering Assessments and Methods (DREAM) international competition which involves simulated generation of datasets

to speed up modelling and evaluation of inferences in regulatory networks [88], [89]. After inference of the transcription networks using the simulated datasets, assessment is done by comparing the inferred network with a benchmark network [90] such the network of the fully sequenced Drosophila Melanogaster and measures such as sensitivity, accuracy, specificity, Matthews Correlation Coefficient and F1-score are used to evaluate the performance of the inference algorithm [84], [91]. The receiver operating characteristics (ROC) curve is usually plotted as sensitivity versus specificity and the area under the curve (AUC) used as an indicative measure of performance where higher number represents better performance [92].

3.1.2 Present Inference Methods

There are various existing methods for inferring gene regulatory networks which have been categorised into differential equation methods and knowledge-based methods. Other methods such Boolean Networks and Correlation-based methods exist but are however beyond the scope of this study.

3.1.2.1 Differential Equation Methods

The formalisms of differential equations such as partial and ordinary differential equations have widely used to study dynamical systems in engineering and scientific research. These powerful methods have been adapted to model and represent metabolic processes and dynamics of gene regulation processes [93], [94]. Here ordinary differential equation (ODE) is focused on as it can be used to simultaneously represent differentiations in time and the dynamics of causal relationships using the earlier discussed four inference levels of Figure 3.1.

In ODE, the rate of change of a gene expression's component is represented as a concentration of all the components, and within the ODE systems, the causal effects of the genes are fixed [95], [96]. ODE can be mathematically formulated as

$$\frac{dX}{dt} = \theta(t, X) \tag{3.1}$$

where $X = X(t) = (X_1(t), ..., X_n(t))^T$ denotes the gene expression values of genes $1, ..., n$ at time point $t, t \in [t_0, T], 0 \leq t_0 \leq T \leq \infty$. The function θ describes the relationship between the first-order derivative of X and how genes are concentrated in the regulatory system. The function could be linear or nonlinear and describes the relationship between the rate of change in concentration of the genes and their causal regulators. For a linear ODE model, the relationship could be written as

$$\frac{dX_i}{dt} = \alpha_{i0} + \sum_{j=1}^{n} \beta_{ij} X_j(t), j = 1, ..., n \tag{3.2}$$

where α_{i0} is the intercept and $\beta = \{\beta_{ij}\}_{i,j=1,...,n}$ represents how the genes in the regulatory system affect the rate of change of expression of the t-th gene.

Reconstructing the regulatory network from data is then transformed as a problem of parameter identification of the ODE system. To identify these parameters, likelihood and least squares-based methods have been used in the past [97], [77]. These methods are however not very effective for reverse engineering and an integrative pipeline approach was proposed [95], [96]. This method involves a two-step process where the first step involves fitting the gene expression's mean curves and estimating its derivative. At the second step, variable selection is carried out using regularization methods such as LASSO [98] and the Smoothly Clipped Absolute Deviation (SCAD) [99] to optimally shrink variables.

Gene regulations in ODE model are modelled and represented by derivative equations where the dependent variables which is the rate of change of one gene's expression, is quantified as a function of other related genes. The TFs regulate the transcriptional processes of the genes in the regulatory system and the independent variables are assumed to be made up of the TF proteins. This assumption enables the reverse engineering of the regulatory network to become a problem of parameter inference of some functions such as linear functions used to model gene expression data [95].

Due to the underlying differences in the statistical and mathematical perspectives of inferring a GRN, ODE aids modelling of the regulatory system but not its inference [100]. This is because functional relationships are assumed to be described by derivation equations then inference of the regulatory architectures is done by statistical methods such as variable selection and parameter estimation [95] and the time delays associated with the self-degradation and activations can integrated into the system by the introduction of respective self-degradation and activation terms into the differential equations.

3.1.2.2 Knowledge-based Methods

Genuine and accurate transcriptional regulations are difficult to identify sorely by reliance on data-driven approaches and the efficiency of reverse engineering using only gene expression data is hard to promise [101], [97], [102]. This makes integration of a priori knowledge of regulations an important requirement in the development of more useful and robust methods of inference. Key functional linkages and relationships can be derived from documented regulations [103], [104], protein-protein interactions [91], protein-DNA binding data from ChIP-Seq [105] and motifs from TF binding sequence [82]. Transcriptional regulatory networks can be identified by the integration of these a priori knowledge with gene expression data.

Integrating a priori knowledge reliably often leads to probabilistic ways of inference such as Bayesian Network [106], [107]. Many techniques have been proposed to calculate these probabilities. In the area of statistical physics an energy function approach was proposed by [108], [109] which involves introduction of prior knowledge from multiple sources by expressing prior knowledge as a function of network energy. Gibbs distribution is used to obtain the prior distribution over the network structures [108]. The weights of the prior knowledge are represented by the parameters of the Gibbs distribution. As a result, the Bayesian network integrates the prior knowledge in order to learn the structure of the regulatory network. The authors of [108] and [109] achieved higher inference performance using both real and simulated data.

[110] proposed a linear programming (LP) method based on ODE for integration of prior knowledge in reverse engineering of GRNs. In the proposed method, the association gap between network structure and gene expression data is minimised by building an LP model and obtaining an integrated regulatory network by solving the LP. The objective function of the LP is then to minimise the number of gene connections needed to have the sparseness of the inferred network with constraints being the prior knowledge and the linear additive equations. The solution that gives the sparser network with respect to the two constraints is chosen as the inferred network [110].

3.2 Modelling and Representing Uncertainty using Bayesian Networks

In real world applications, time is of great importance as events in all fields of life happen in time. In medical diagnosis, the condition of a patient with HIV may be observed to improve with time given that the patient is on regular dosage of some anti-retroviral

drugs. Accurate representation of uncertainty is therefore crucial in biomedical applications. Before exploring what temporal events are and how they are modelled, static modelling using Bayesian network is introduced with a few application examples.

A Bayesian Network (BN) is a Directed Acyclic Graph DAG having nodes that represent random variables which may be discrete or continuous [13],[111] which can be a set of finite variables from $x_1, ..., x_n$, where x represents features to be modelled. A set of directed links or arrows connect pairs of nodes in a graph G. If there is an arrow from node x to node y, x is said to be the parent of y. Each node x_i has a conditional probability distribution $P(x_i|Parent(x_i))$ that qualifies the effect of the parent on the node.

$$Px_1, ..., x_n = \prod_{i=1}^{n} Px_i|parents(x_i) \tag{3.3}$$

Figure 3.2 shows a simple Bayesian network that illustrates the relationship between a student's intelligence, grade and the score in the Scholastic Assessment Test (SAT).

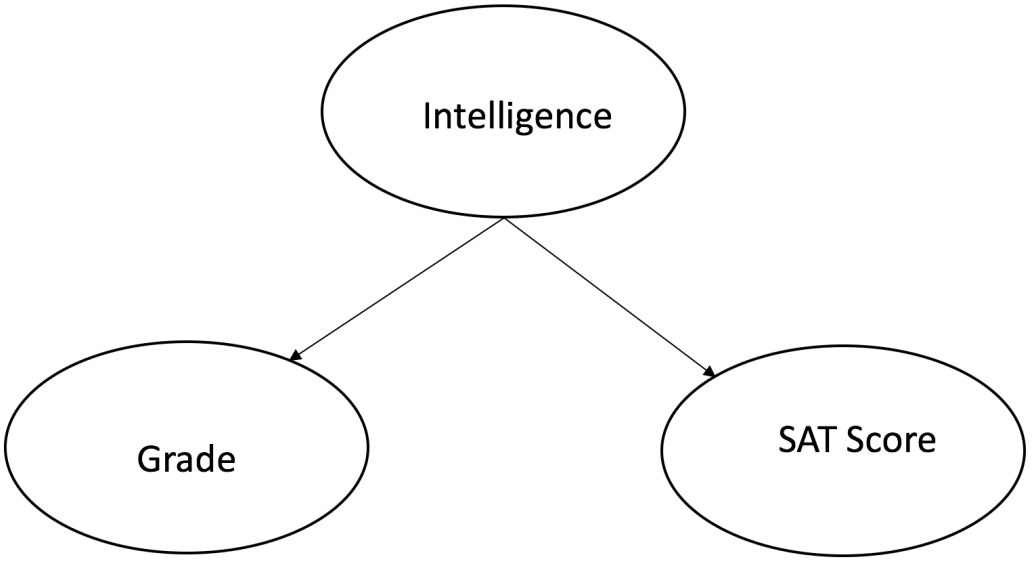

FIGURE 3.2: A Simple Bayesian Network showing how a students intelligence affects grade and score in the Scholastic Assessment Test (SAT)

Bayesian Networks have been applied to many real world applications such as modelling cognitive behaviours [112], variable optimisation of medical images [113] and various medical prognosis [114] and protein predictions [115]. The major advantage of BN is that they contain no redundant values of probability and hence have no chance of inconsistency [13], [116]. BN algorithms are more capable of handling very noisy biological experimental data due to their probabilistic nature [117], [118]. They can more easily handle incomplete datasets since they probabilistically encode correlations between input variables [119]. Another advantage of BN is that they allow construction of causal relationships and determination of influences in expression data variables [119], [117] however, Bansal et al [120] argued that BNs strictly define only probabilistic dependencies between variables and therefore causal links cannot be assumed except if the causal Markov condition is true.

Despite the advantages of BNs, they can only be used to model static events and do not take into account temporal event that happen over a period of time [121], [122] and modelling cyclic phenomena such a feedback loops and cyclic regulations are not possible with BNs [123]. These limitations are overcome by dynamic Bayesian networks (DBNs) which are extensions of BNs that incorporate time dependencies and have been proposed by various authors [124], [125], [126], [127], [128], [129].

3.3 Representing Uncertainty in time: The DBN

Temporal models are processes where there are replications over time. They are used to represent uncertainty in events that happen over time. There are three main kinds of temporal models commonly used. These are Hidden Markov Models, Kalman Filters and Dynamic Bayesian Networks (DBNs). In bioinformatics, DBNs are used to model gene regulatory networks and infer temporal relationships between features at

successive time points. A popular DBN framework used for inferring gene regulatory network is representing it as a vector autoregressive models of order 1 [130], [1], [131], [25].

The temporal models are also referred to as state-space model which is a term used to imply that the hidden state is a vector [132]. Temporal events can be seen as series of snapshots or time slices containing a set of random variables. Time is usually discretised using time granularity. This time is the interval at which measurements can be obtained from a sensor [14]. In representing their probability distributions for state-space models, the Markov assumption is usually made. The Markov assumption states that the state of the next time point is independent of the past state given the present state [133], [14], [134].

DBNs identify time-course regulatory influences in biomedical data on the framework of BNs where time is modelled as a stochastic temporal process over time series set of random variables.

Figure 3.4 shows graphical representation of a network containing cyclic regulations with cycle $X1 \rightarrow X2 \rightarrow X5 \rightarrow X1$. This kind of network cannot be modelled by a BN however, a DBN models this cyclic regulation and feedback loop common in biological systems by dividing the states of the variables by time points as shown on the right. Assuming each time point were as a single variable, the simplest causal network for a time series sequence of data would be modelled as a first order Markov

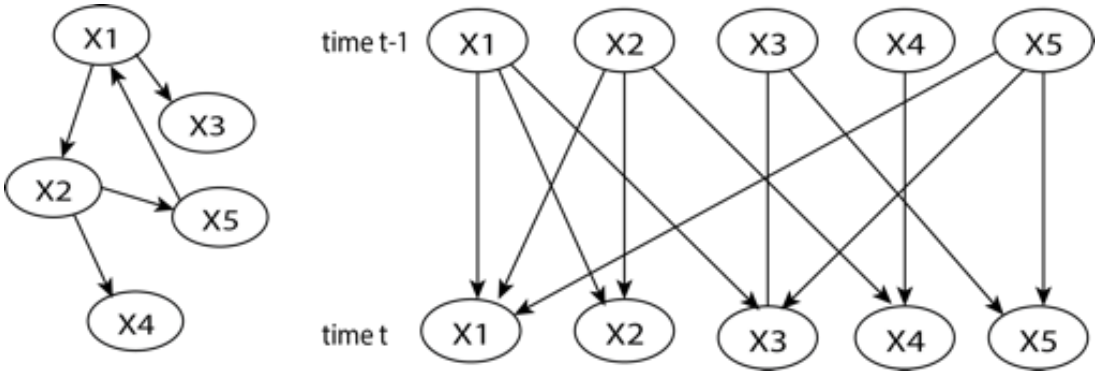

FIGURE 3.3: Graphical representation of a network with cyclic regulations.

chain where the state of the next variable is dependent only on the state of the previous variable [13].

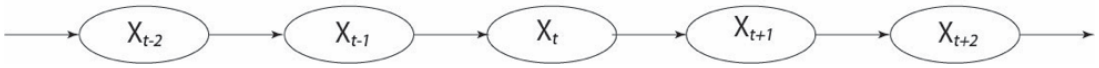

FIGURE 3.4: A DBN with sequence of time points corresponding to first order Markov chain

Figure 3.4 shows a DBN corresponding to a first order Markov chain where X_{t+2} depends only on X_{t+1} and X_{t+1} depends only on X_t. X_t then depends only on X_{t-1} and X_{t-1} depends only on X_{t-2}. Dependencies over more than one time step between variables cannot be represented by the Markov chain [135]. A way to however extend the model is to assume that there are hidden discrete variable known as states on which the observed variables depend on. This sequence of hidden states are known popularly as Hidden Markov Models (HMMs).

A Hidden Markov Model is a temporal probabilistic model in which a single discrete random variable is used to describe the state of a process and stochastic processes are represented as Markov chains where the states are not directly observed [136].

Figure 3.5 shows a HMM representation where $Z_1, Z_2, ..., Z_n$ represent some hidden discrete random variables and $X_1, X_2, ..., X_n$ represent some observed random variables.

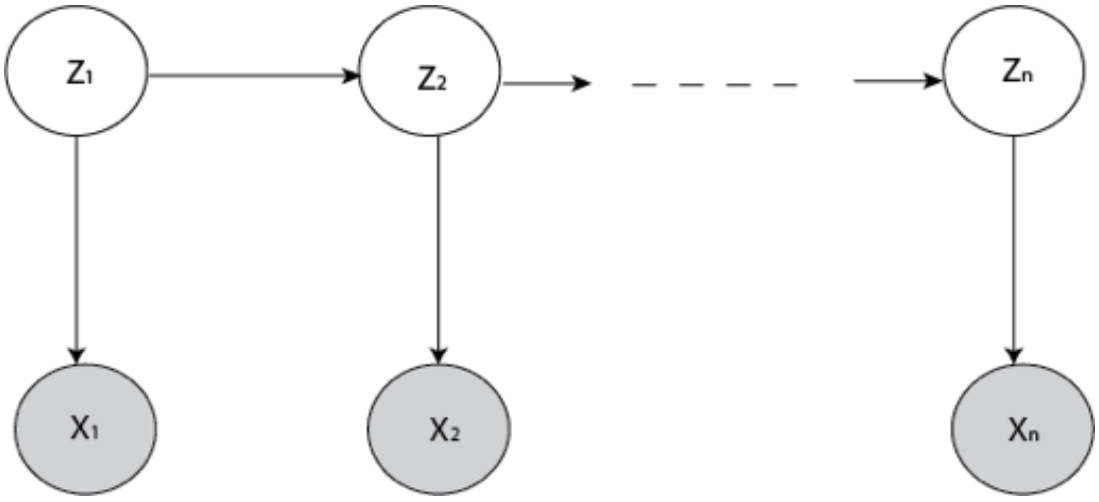

FIGURE 3.5: A HMM showing Hidden variables $Z_1, Z_2, ..., Z_n$ and observed variables $X_1, X2, ..., X_n$

Hidden Markov Models have been used in many real world applications such as speech recognition [137] and classifying speech response of the brain [138]. HMMs have further been combined with Bayesian networks to address the problem of overfitting and model regularization for speech recognition [139], [140]. HMMs have been used to model surgical performance and expert surgical gesture by processing kinematics data such as movement of surgical instruments in time [141]

Despite the application capabilities of HMM and it being the simplest form of DBN [135], there are inherent problems which make its application limited in some complex real world applications [142],[132], [143], [13]. First, the initial states in HMMs are represented by only single discrete random variables and are not suitable for representing temporal uncertainties with two or more state variables [142]. Another problem with HMMs is the computational complexity in tasks such as tracking objects in a sequence of images. For instance, if there are 20 Boolean state variables in a network each with three parents, the HMM will have 2^{20} states and therefore 2^{40}, almost a trillion, probabilities in the transition matrix. This large transition matrix makes the inference of HMM more computationally expensive and the problem of learning such large number of parameters makes it unsuitable for high-dimensional data such as gene expression data and other large biomedical data. A corresponding DBN transition model for the same problem described above will have $20 \times 2^3 = 160$ probabilities [13], [142] hence DBN is preferred over HMM for modelling temporal uncertainty in bioinformatics.

Kalman filter models (KFMs) can be seen as HMMs with conditional linear Gaussian distribution and also a simple form of DBN [144]. They are Bayes optimal minimum mean-squared error estimator for linear systems with Gaussian noise [145]. Since Kalman filters are recursive data processing algorithms for solving discrete linear filtering problems, the estimates of the system variables are updated at each time step using only the observations at that time without having to store all past observations [145]. Kalman filters are powerful in solving localization estimation problems [146]

and have been used for object tracking [147], robot control [148] and 3-D modelling [149]. KFMs have also been applied in statistical neuroscience for spike sorting [150] and modelling physiological tremor [151]. The assumption of a linear Gaussian model is however too strong and an extended Kalman filter (EKF) was proposed to overcome the problem of nonlinearities [152]. The EKF has been applied to neural networks training and to data fusion problems [153] and for moment estimation in bimolecular reactions [154]. However authors of [155] argued that the series of approximations in EKF algorithm can lead to the nonlinear functions and associated probabilities being poorly represented. Another problem is that large qualitative discrete variables cannot be represented since the system is assumed to be jointly Gaussian [13]. For this reason, the more efficient DBN model is preferred in this study.

3.4 Machine Learning in Bionformatics

Machine learning is a multidisciplinary field of study that involves computer science, statistics, artificial intelligence and programming. Machine learning is the ability of computer systems to gain or acquire knowledge about a task without further explicit programming. Tom Mitchell [156] defined machine learning as ”A well-posed learning problem: A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E.” The field of machine learning seeks to answer the question of how a computer system can be built to automatically learn from experience and take actions based on what has been learned.

Machine learning algorithms are broadly divided into supervised learning, unsupervised learning and reinforcement learning [157]. Supervised learning involves presenting an algorithm with example inputs and desired outputs called training data. The goal of the learning algorithm is to map inputs to outputs [158]. There are two main supervised learning tasks —classification and regression.

Classification problems involve identifying which label a set of training features belong to. An example is trying to determine whether an email is either spam or not or whether a patient has cancer or not given gene expression profiles. Regression involves predicting continuous valued outputs (rather than discrete as in classification). The task is to estimate relationships between one or more independent variables and a dependent variable.

In unsupervised learning, there are no labels given to the learning algorithm. The algorithm is left to find and discover hidden patterns in the data. For example finding different groups and pattern in relationships given a gene expression data using methods such as clustering and algorithms such as K-Means Clustering and Hierarchical Clustering. Patterns such as co-expression in genes can be discovered in high-dimensional data using clustering [159].

In reinforcement learning, an algorithm tries to find suitable actions to take in a given situation in order to maximise reward. For example an algorithm tries to perform task such as driving a car without being given examples of optimal outputs (as in the case of supervised learning) but must through trial and error discover them [134]. Reinforcement learning has been used to model reward prediction errors for patients with schizophrenia where the goal was to assess belief formation and belief perseveration [160]. Zhu et al also used reinforcement learning in addition to text mining to construct protein-protein interaction network for diagnosis of prostrate cancer [161]. Reinforcement learning continues to grow as an active area of research but it is however beyond the scope of this study.

3.5 Application Domains of Machine Learning for Disease Prognosis

The core cellular molecules in bioinformatics include DNA, RNA, Proteins and metabolites. The central dogma of molecular biology is the transcription of DNA into RNA and the translation of RNA into proteins. Some proteins may act as catalysts for production of metabolites. Studying these interactions is one of the main focus of research in bioinformatics however faster and cheaper computing power have increased the possibilities of discovering novel interactions among these molecules.

Machine Learning methods have been widely successful in bioinformatics. With current high-throughput gene expression data of high volume, variety and velocity, machine learning algorithms have helped speed up analysis that would have had long completion times. Understanding bioinformatics data such as proteomics, genomic, transcriptomic and metabolomic data have widely presented research problem. The problem of mining and understanding these complex data is even increased due to there high dimension where the number of features is in thousands. This requires complex learning algorithms and in many cases, a combination of algorithms and ensemble methods.

Support Vector Machines (SVMs) are supervised learning techniques for classification and regression analysis tasks [26]. They are used to analyse and recognise patterns in data. For a given set of training data, the SVM builds a model by constructing a hyperplane (decision boundary) that is used to mark the data as belonging to one of two categories. The SVM can also be used to perform non-linear classification by implicitly mapping data inputs into high-dimensional feature space using kernel dot product. The SVM was especially made popular in systems biology when it was successfully applied with the recursive feature elimination algorithm by Guyon et al [28] to select genes relevant for cancer classification achieving high accuracy of 98%. Kara et

al [162] applied SVM for the prediction of protein-protein interactions in which the authors developed a web server aimed at interfacing a sequence-based predictor for predicting pairing of histidine kinase.

Owens et al [163] applied SVM for early detection of ovarian cancer. The SVM achieved a diagnostic accuracy of 74% for Raman spectra of blood plasma and an accuracy of 93% for the Attenuated Total Reflection Fourier Transform Infrared (ATR-FTIR) spectra of blood samples. They concluded that with SVM and biospectroscopy, spectral alterations associated with ovarian cancer could be identified. Wang et al [164] used SVM to detect four genes which accurately differentiated between benign and malignant thyroid nodules. The genes namely Fibronectin 1 (FN1), neuronal guanine nucleotide exchange factor (NGEF), gamma-aminobutyric acid type A receptor beta 2 subunit (GABRB2) and a high mobility group AT-hook 2 (HMGA2) all had overall performance of 97% and 93.8% sensitivity and specificity respectively.

The problem of accurate feature selection have been generally known and widely studied since the early days of machine learning [165]. [28] believed that a criterion for good feature ranking was not necessarily a criterion for good feature subset ranking and went on to develop the now popular Recursive Feature Elimination (RFE) algorithm. The algorithm works in three main steps:

- Train the classifier such as the SVM to optimise the weights.
- Compute the ranking criterion for all features
- Remove the feature with the smallest ranking criterion.

Application of the RFE algorithm using the weight magnitude of the SVM yields the SVMRFE algorithm. The SVMRFE algorithm in more detail can be represented thus.

Inputs:

Training examples

$X_0 = [x_1, x_2, ..., x_n]^T$

Class labels

$y = [y_1, y_2, ..., y_n]^T$

Initialize:

Subset of surviving features $s = [1, 2, ...p]$

Feature ranked list

$r = []$

Repeat until $s = []$ where n is the number of examples and p is the number of predictor features.

Experimental results on two gene expression datasets (colon cancer and leukemia) obtained from DNA micro-arrays [166] showed that when the SVMRFE is applied, linear kernel performed best, followed by the radial kernel and lastly the polynomial kernel. Higher order polynomials are computationally expensive and do not generalise well especially in high-dimensions. A large value of SVM cost of constraint C results in low bias and high variance while a small value of C results in high bias and low variance. For the radial basis function (RBF) kernel, large σ^2 which is inversely related to γ results in higher bias and lower variance while a small value of σ^2 results in lower bias and higher variance. The linear kernel generally performs better for situations where $p >> n$ due to less overfitting because of the small number of examples and less complexity while for $p << n$ scenarios, the radial kernel with a more complex decision boundary performs better.

Figure 3.6 shows the performance of the three SVMRFE kernels used in this study on the colon cancer gene expression datasets as studied by [28]. It contained 62 tissues and 2000 gene expression values. Out of the 62 tissues, there were 40 colon cancer tissues and 20 normal ones. As shown from the figure, linear kernel was most accurate in distinguishing cancerous from non-cancerous tissues followed by the RBF kernel

and lastly the polynomial kernel. The accuracy of the feature subsets were measure in powers of two until the total number of features. The best results for all kernels were achieved at the selection of 16 subsets after which the accuracy of the polynomial kernel begins to go down followed by the RBF kernel. The linear kernel maintained a high accuracy of 100% and then started going down from 512 subsets as more subsets were been selected. All of the kernels showed downward trends as the number of features increased.

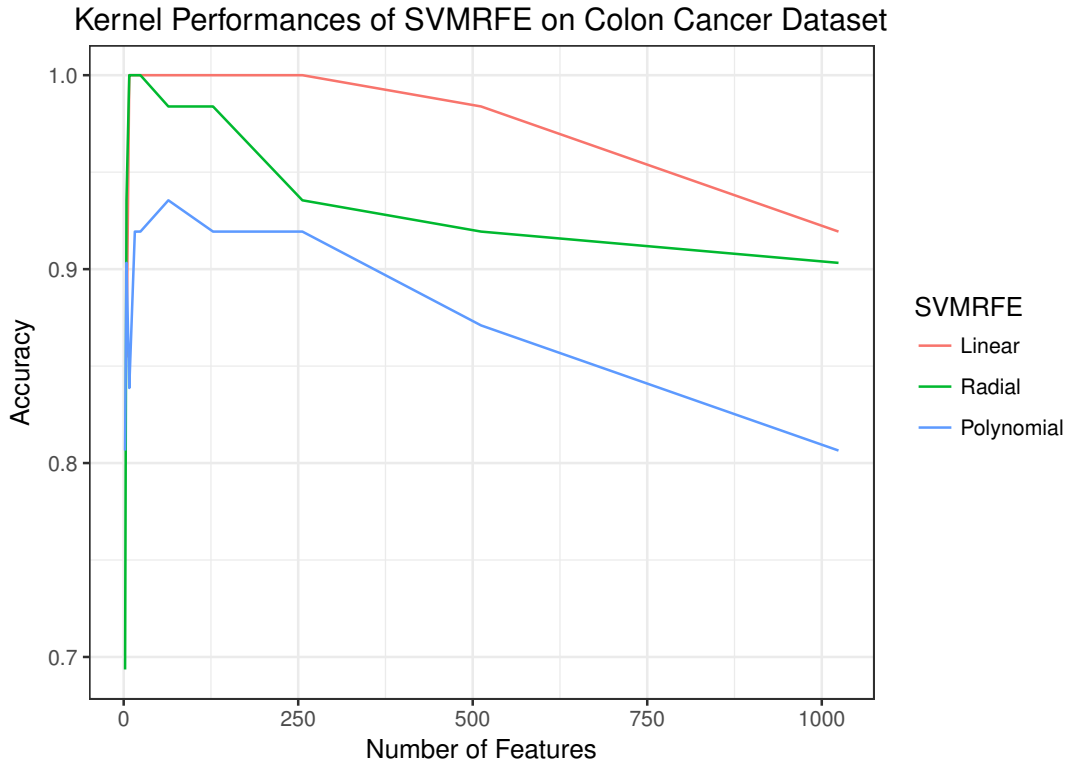

FIGURE 3.6: Performance of various SVMRFE kernels on Colon Cancer Dataset

Figure 3.7 shows the performance of the SVMRFE on a larger leukemia dataset. The problem here was classifying two different types of leukemia (ALL and AML). The dataset consists of 72 samples and 7129 features. The 72 samples were made up of 47 ALL and 25 AML samples. The figure shows a similar trend with the results from the colon cancer experiment. The linear kernel had the best performance followed by the RBF and lastly the polynomial kernel. The kernels had their peak performance at 64 subsets when the polynomial reached its best performance of 98.61%. Even though this was slightly better than the best performance of the RBF at 97.22%, the RBF had

overall better performance than the polynomial kernel which saw its performance go down to 75% when 2048 subsets were selected while the linear was still at 100%.

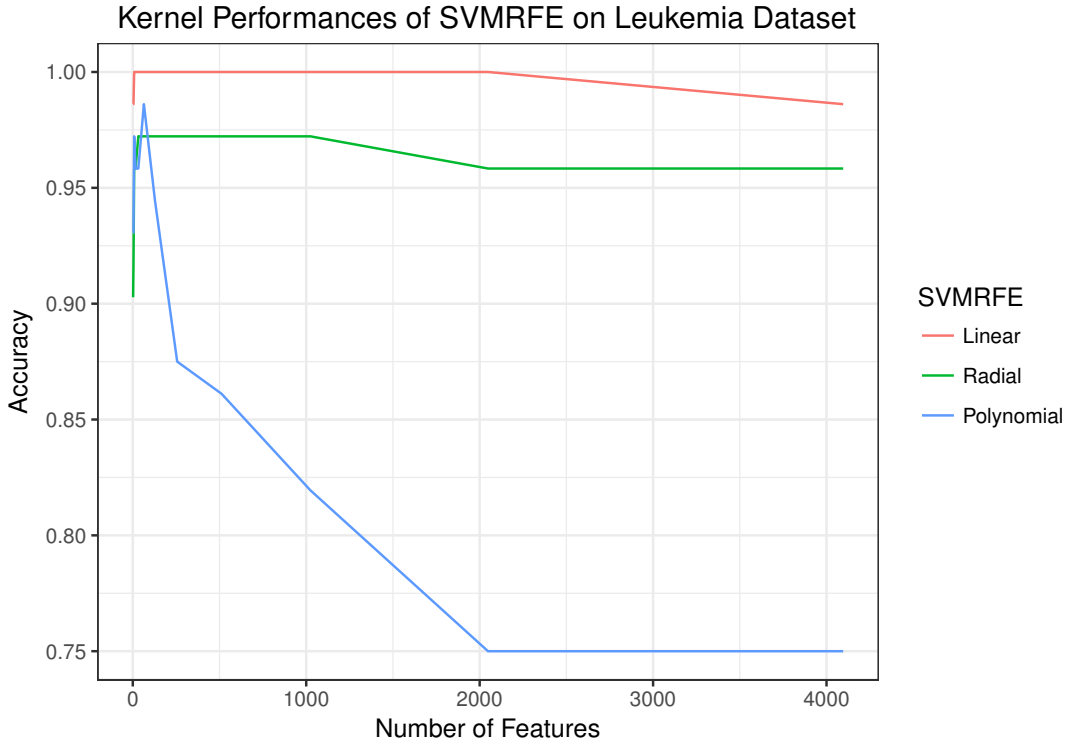

FIGURE 3.7: Performance of various SVMRFE kernels on Leukemia Dataset

3.6 Conclusions

The chapter reviews relevant literature and current methods for reverse engineering of gene regulatory networks from gene expression data. It discusses the generic reverse engineering framework and present existing methods which include differential equation and knowledge-based methods. It explores relevant literature involving machine learning in bioinformatics such as Support Vector Machine and its adaptation in the diagnosis of cancer.

It describes Bayesian Network and uncertainty representation in time using Dynamic Bayesian Network and alternative methods such as the Hidden Markov Models. It also shows the three kinds of SVMRFE kernels used in this study: the linear, the RBF and

the polynomial kernels and how their cores vary with the number of feature subsets in a classification task. Results from two datasets from the study of [28] are shown. The kernels all generally show a downward trend as the number of features increase with the linear kernel being better for $p >> n$ scenarios.

The gaps discovered are that DBN is more suitable for modelling and inferring relationships among variable at different time points that the use of KFMs and HMMs and hence the choice of adoption in this thesis. While machine learning techniques such as SVMs are good at selecting high quality features, ways of inferring the relationships between these selected features if each observation is a time point have not yet been studied.

The linear assumptions of the DBN representation of a vector autoregressive process (VAR) process is too strong and may not capture the true representation of complex biological systems known to be inherently nonlinear. Nonlinear methods have to be developed for more efficient modelling and representation of DBNs. This may potentially yield novel relationships and interactions that may help clinicians in understanding the nature of disease metastasis and in discovery of new drugs. For new drugs to be developed for various diseases, genomic, proteomic and metabolomic data about them have to be analysed using computational methods. These kinds of data were used in this thesis and chapter 4 seeks to describe them.

Chapter 4

Description of Datasets

4.1 Introduction

In chapter 3 relevant literature and current methods for reverse engineering of gene regulatory networks from gene expression data were discussed. It explained the generic reverse engineering framework and presented existing methods which include differential equation and knowledge-based methods. Bayesian Network was introduced and representation of uncertainty in time using Dynamic Bayesian Network explored.

This chapter explains the different key biomedical data used in this thesis to show robustness of methods developed. It studies ovarian cancer metabolite data, hypertension gene expression profiles data, colorectal cancer protein profiles data, ovarian cancer time-course data. It introduces the Dialogue on Reverse Engineering Assessment and Methods (DREAM) datasets, the Escherichia coli and the Drosophila Melanogaster datasets. All data used in this thesis are anonymised and publicly available. Ethical approval was granted on the 25th of April 2013 prior to the commencement of the experiments.

4.2 Ovarian Cancer Metabolite Dataset

The dataset was generated by the study conducted Guan et al [19]. Serum samples were obtained from the Ovarian Cancer Institute (OCI) Atlanta, GA and consists of two study groups, 37 patients with papillary serous ovarian cancer and 35 control groups. The mean age of the patient group was 60 years with age range of 43-79 years all having stages I-IV of the disease. The mean age of the control group was 54 years with age range of 32-84 years.

The samples were obtained after approval by the Institutional Review Board. The cancer patients and control group were required to avoid food, medicine and alcohol for 12 hours before sample collection. 5ml of blood samples were subsequently collected from each donor by venipuncture. The samples were centrifuged at a low temperature of 4 degrees Celsius for 5 minutes at 5000 r.p.m and the serum obtained subsequently.

Due to the complex nature of the serum samples, adduct ion analysis was performed to assign signals to the mass spectrum. Adducts that usually form in the positive ion mode electrospray (ESI) include $[M+H]^+$, $[M+NH_4]^+$, $[M+N_a]^+$, $[M+K]^+$, $[M-H_20+H]^+$ and $[2M+H]^+$ species. For the negative ion mode ESI, they include $[M-H]^-$, $[M+CH_3COO]^-$, $[M+Cl]^-$, $[M+HCOO]^-$ and $[2M-H]^-$.

Support Vector Machine Recursive Feature Elimination (SVMRFE) methods were used to select features. These are SVMRFE with linear and non-linear kernels, L1-norm SVM and the Weston feature selection using SVM non linear kernel (SVMRW). Further investigation was performed with MzMine software [19] using Liquid Chromatography Time-of-Flight Mass Spectrometry (LC/TOF MS) which showed that a total of 576 positive ion mode and 280 negative ion mode features were extracted.

Stratified analysis of the highly complex data revealed presence of redundant species such as dimers, adducts and isotopes which were removed. The final resulting dataset comprise of 360 features in positive ion mode and 232 features in negative ion mode

making a total of 592 metabolomic features from 72 observations (37 cancer patients and 35 control volunteers) used in this study.

4.3 Hypertension Gene Expression Profile Dataset

The gene expression profile dataset of hypertensive patients used in this study was generated from the works of Lynn et al [20]. It consists of expression profiles of male young-onset hypertension. After approval by the Institutional Review Board of Academia Sinica, the highest academic institution in Taiwan, participants were recruited from MJ life enterprise company healthcare facility with informed consent. The recruitment criteria for the participants included: (i) Body Mass Index (BMI) $<$ 35; (ii) age range 20-50 years; (iii) at least 8 hours of fasting; (iv) fasting blood sugar $<$ 126 mg/dl; (v) no prior history of cancer or other liver, heart, lung or kidney diseases; (vi) not on any hypertension medication and (vii) without major symptoms of acute hypertension in the previous two weeks.

Participants' blood pressure were measured three times and the average of the last two measurements was used for the diagnosis of the disease. Participants were classed as hypertensive if their systolic and diastolic blood pressures were greater than 140mmHg and 90mmHg respectively, and normotensive otherwise. A total of 77 male hypertensive patients with median age of [37.6 +- 7.2] and 82 normotensive controls of median age [36 +- 6.6] took part in the study resulting in 159 total participants.

10ml of blood was obtained from each patient and frozen immediately at 70 degrees Celsius and total RNA was subsequently extracted from the blood using the Human OneArray microarray platform by the Phalanx Biotech Group, Taiwan. Four copies of data were generated for each participant. There were a total of 39200 polynucleotide data in each microarray chip of which 22184 were eventually mapped to the human genome.

Various quality control parameters were used for the microarray data. These include coefficient of variation, Pearson correlation coefficient, percentage of present calls and array quality filter. Global data normalization using logarithm and z-score was carried out and a total of 103 out of the 22184 genes were found to be differentially expressed with p-values less than 0.01 for both hypertensive and normotensive groups. In summary the dataset used had 22184 features and 159 observations (rows).

4.4 Colorectal Cancer Protein Profiles Dataset

Colorectal cancer dataset was generated by the study of De Noo et al [21]. Serum samples of 66 colorectal patients were obtained a day before surgery. The patients were at stage IV of the disease and the extent of the tumour spread was determined by the Tumour Node Metastasis (TNM) classification. The median age of the patients was 62.8 years with a range of 32.6-90.3 and male to female ratio of 31:35. The control group was made up of 50 healthy volunteers with median age of 49.7 years ranging from 25.9-76.6 and a male to female to ratio of 21:29.

After approval by the medical ethics committee, 10cc of blood samples were drawn following informed consent of the participants while they were seated and non fasting. Serum was obtained after centrifugation at 3000 r.p.m for 10 minutes and stored at -70 degrees Celcius. Samples from the 116 participants were distributed randomly across three plates in a randomized block design [167], [168] of roughly equal equal proportions. Analysis was carried out for three consecutive days.

Isolation of the peptides from the protein was done using magnetic beads based hydrophobic interaction chromatography (MB-HIC) kit from Bruker Daltonics (Bremen, Germany) following manufacturers instructions. α-cyano-4 hydroxicinnamic acid (0.3g/l in ethanol:actone 2:1) was used as matrix. The preparation was done on an

8-channel Hamilton STAR pipetting robot (Hamilton, Martinsried, Germany). Matrix assisted laser desorption ionisation time-of-flight (MALDI-TOF) mass spectrometry was carried out using an UltraFlex TOF/TOF instrument from Bruker Daltonics. Ion-formed N2 pulse laser beam were accelerated to 25kv. With this specifications, peptide/protein peaks in the range of 960-11,169 Da were measured.

All unprocessed spectra were exported in binary ASCII format and they consisted of approximately 65,400 mass-to-charge ratio (m/z) values. Further low level analysis were performed by [169] using Bruker Daltonics ClinProTools software version 2.2. Recalibration and top-hat baseline correction using default parameters was performed. Outlier detection with data reduction factor of 4 was done and two additional outliers were detected from the control group. MATLAB software was used for interpolation of all cancer and control spectra which resulted in a dataset of 64 cancer and 48 control observations with 16331 spectral features used in this study.

4.5 Time-Course Ovarian Cancer dataset

Time-course ovarian cancer dataset was generated from the works of Yseult et al [22] using A2780 human ovarian carcinoma. IC_{90} concentrations of oxaplatin (32μm) or cisplatin (25μm) was introduced to the samples for 2 hours and were then allowed to grow in a drug-free environment during the time-course experiment. The cells were processed in three stages; before treatment, immediately after treatment 0hrs, and at 2hrs, 6hrs, 16hrs and 24hrs after treatment. Similar experiments were done for the control experiment but cells were however exposed to drug-free medium.

The quality of each RNA sample during each step of the process was evaluated using gel electrophoresis and spectrophotometry. The RNA samples were purified and fragmented and each fragment sample was carefully assessed by hybridization to Affymetrix HGU95Av2 chips. The expression summaries of the probe levels were

obtained using Robust Multichip Average (RMA) method 43 which involved the use of a linear model fitted to quantile normalized and log transformed probe intensities. The RMA calculations were done using a Bioconductor function justRMA available in affy1.4.32 of the Bioconductor project 1.4 package.

The expression levels of each gene was modelled as a function of time having an initial state response followed by an exponential increase or decrease to an asymptotic final response. Genes were selected from the dataset which initially included 12625 gene probesets. From these probesets 34 genes were finally selected based on three selection criteria. The first criterion involved selection of genes whose expression levels were increased or decreased by both cisplatin and oxaliplatin platinum drugs.

The second selection criterion involved selection of genes differentially affected by both the cisplatin and oxaliplatin. Genes selected by the second criterion include 3 genes with no material changes in oxaliplatin versus control samples but with increase in cisplatin versus control; 1 gene with no significant changes in oxaliplatin versus control but with significant decrease in cisplatin versus control; 6 genes with no material changes in cisplatin versus control but with increase in oxaliplatin versus control and 10 genes with no material changes in cisplatin versus control but with decrease in oxaliplatin versus control.

After the selection criteria of both the first and the second have been met, the third and final criteria involved visual inspection of time-course and concentrated-effect patterns. A total of 34 genes were selected as follows: 3 genes with no changes due to oxaliplatin but increase due to cisplatin; 0 genes with changes due to oxaliplatin but increase caused by cisplatin; 5 genes with no changes caused by cisplatin but with significant increase by oxaliplatin; 9 genes with no changes due to cisplatin but with decrease by oxaliplatin; 12 genes with increase caused by both drugs and 5 genes with decrease caused by both drugs.

4.6 Reverse Engineering Datasets

This section explains the both simulated and real world reverse engineering datasets used in this study. These are benchmark datasets commonly used to validate developed algorithms for reverse engineering of gene regulatory networks from time-course gene expression data.

4.6.1 DREAM Datasets

The Dialogue on Reverse Engineering Assessment and Methods (DREAM) initiative organises reverse engineering competitions annually with the goal of predicting the connectivities of networks from both simulated (*in silico*) and real (*in vivo*) gene expression datasets. Two DREAM datasets were used in chapter 7 of this thesis: DREAM3 dataset comprising of three Yeast knock-out expression matrix of sizes 10, 50 and 100 nodes [67] and DREAM4 dataset comprising of five Yeast datasets each of 10 nodes and 21 time points [68]. First 10 features of the 100 node network of DREAM3 dataset are shown in Appendix A. The *in silico* networks of the DREAM datasets are generated by differential equation. The networks were modelled as subgraphs of real sequenced *S. cerevisiae* (Yeast) and *E. coli* gene expression networks [170]. The DREAM datasets are publicly available from the GeneNetWeaver website [171].

4.6.2 Escherichia coli Dataset

Miroslav Radman [172] carried out the first experiment that supported the existence of inducible DNA repair network in *Escherichia coli* popularly known as *E.coli*. He used the term ”SOS response” to describe the network in which during regulation of the network, two proteins play key roles. The first protein, LexA acts as a repressor while the second protein, RecA, acts as an inducer. The repressor binds to

specific sequence —the SOS box which is present in the promoter region of the SOS genes during normal growth and prevents their expression. The amount of repression depends on the exact sequence of their SOS box. The amount of LexA repressor after ultraviolet light irradiation decreases about 10 times in a few minutes however, the genes are not all induced at the same time. Genes *uvrA*, *uvrB* and *uvrD* are the first to be induced. These together with endonuclease *UvrC*, catalyse nucleotide excision repair (NER) which excises the damaged nucleotide from double-stranded DNA [173]. The 9 square matrix time-course SOS DNA repair network used in this thesis was published by [174] and is freely available from their website https://github.com/xiangdiuxiu/NetworkDC/tree/master/data/SOS and also shown in Appendix A.

4.6.3 Drosophila Melanogaster Dataset

Drosophila Melanogaster also known as fruit fly has been widely used in biological research in the areas of genetic, microbial pathogenesis, physiology and life history evolution. Authors and scientists of [175] determined the nucleotide sequence of approximately 120 megabase euchromatic portion of the Drosophila genome. They used a whole-genome shotgun strategy supported by a high-quality bacterial artificial chromosome physical map and extensive clone-based sequence. The genome encodes approximately 13,600 genes. The dataset is publicly available at fly base database http://flybase.org/. Appendix A shows the subset of 67 genes and 11 time points used in this study to compare the efficiency of developed algorithm which is publicly available from the author's website http://icube-bfo.unistra.fr/en/index.php/G1DBNpractice [131].

4.7 Conclusion

This chapter describes the datasets used in this thesis. All the data used are anonymised and publicly available. Ethical approval for the research was granted on the 25th of April 2013 before the commencement of the experiments. Datasets introduced and explored are Ovarian Cancer Metabolite Dataset, Hypertension Gene Expression Profile Dataset, Colorectal Cancer Protein Profiles Dataset, Time-Course Ovarian Cancer dataset, The Dialogue on Reverse Engineering Assessment and Methods (DREAM) Datasets, Escherichia coli Dataset and the Drosophila Melanogaster Dataset.

In summary the ovarian cancer metabolite data comprise of 592 metabolomic features and a total of 72 observations made up of 37 cancer patients and 35 control volunteers. The hypertension dataset had a total of 22184 gene expression profiles and 159 observations made up of 77 hypertension and 82 control volunteers. The colorectal cancer protein profiles had a total of 16331 spectral features and 112 observations which comprise of 64 cancer patients and 48 control volunteers.

The time-course ovarian cancer dataset had a total of 12625 genes and two drug types: the cisplatin and oxaliplatin. Out of these, 34 were selected for this study based on three selection criteria described in section 4.5. Figure 4.1 shows the 34 ovarian cancer time-course genes genes modelled in this thesis. Reverse engineering datasets are also presented. These include both simulated Dialogue on Reverse Engineering Assessment and Methods (DREAM) datasets and real Escherichia coli and Drosophila Melanogaster datasets. All the datasets are publicly available and can be downloaded from the authors' references and url links provided.

Probeset	Drug Category	Gene Symbol	Gene Title
37842_at	cisplatin increases, oxaliplatin constant	MDFIC	MyoD family inhibitor domain containing
33334_at		ACYP1	acylphosphatase 1, erythrocyte (common) type
39742_at		TANK	TRAF family member-associated NFÎB activator
39633_at	oxaliplatin decreases, cisplatin constant	S100A3	S100 calcium binding protein A3
38287_at		PSMB9	proteasome (prosome, macropain) subunit,
38576_at		HIST1H2	histone 1, H2bd
32805_at		AKR1C1	aldo-keto reductase family 1, member C1
34678_at		FER1L3	fer-1-like 3, myoferlin (C. elegans)
1536_at	oxaliplatin increases, cisplatin constant	CDC6	CDC6 cell division cycle 6 homolog (S. cerevisiae)
39269_at		RFC3	replication factor C (activator 1) 3, 38 kDa
572_at		TTK	TTK protein kinase
40726_at		KIF11	kinesin family member 11
1801_at		BARD1	BRCA1 associated RING domain 1
1515_at		FEN1	flap structure-specific endonuclease 1
41569_at		DNAJC9	DnaJ (Hsp40) homolog, subfamily C, member 9
2086_s_at		TYRO3	TYRO3 protein tyrosine kinase
39631_at		EMP2	epithelial membrane protein 2
37374_at	both increase	ANXA4	annexin A4
36634_at		BTG2	BTG family, member 2
2031_s_at		CDKN1A	cyclin-dependent kinase inhibitor 1A (p21, Cip1)
37579_at		CYFIP2	cytoplasmic FMR1 interacting protein 2
1243_at		DDB2	damage-specific DNA binding protein 2, 48 kDa
40336_at		FDXR	ferredoxin reductase
41191_at		KIAA0992	palladin
1890_at		PLAB	prostate differentiation factor
1173_g_at		SAT	Spermidine/Spermine N1-Acetyltransferase, Alt. Splice 2
33322_i_at		SFN	stratifin
39708_at		STAT3	signal transducer and activator of transcription 3
36079_at		TP53I3	tumor protein p53 inducible protein 3
1945_at	both decrease	CCNB1	cyclin B1
527_at		CENPA	centromere protein A, 17kDa
40117_at		MCM6	MCM6 minichromosome maintenance deficient 6
37228_at		PLK	polo-like kinase (Drosophila)
40619_at		UBE2S	ubiquitin carrier protein

FIGURE 4.1: The 34 Ovarian cancer Genes Modelled in this study.

Chapter 5

Inferring Gene Regulatory Network using a Two-Stage Approach

5.1 Introduction

Machine learning and other computational approaches have been applied in the prediction of biomarkers. This chapter investigates ovarian cancer and hypertension diseases. Algorithms such as Vector Machines (SVMs) [163] and Artificial Neural Networks (ANN) [176] have been applied for automatic detection of ovarian cancer biomarkers. Attempts have also been made in detecting potential biomarkers for hypertension disease using algorithms such as C4.5 classification algorithm [177] and Neuro-Fuzzy systems [178]. These methods demonstrate classification abilities of the algorithms but are not do not consider possible temporal associations of the biomarkers. Efficient algorithms and techniques are needed to discover possible associations between potential biomarkers which may lead to possible discovery of treatment drugs.

Figure 5.1 shows block diagram of the computational model representing the two-stage approach adapted in this chapter. The model takes biomedical data as input and

applies feature selection on the data to determine best performing features. In this chapter, two case studies were analysed using their respective datasets. The first involves ovarian cancer datasets and the second involves hypertension datasets. Feature selection was carried out at the first stage using different methods, more specifically, RFRFE, LASSO, and SVMRFE with linear and RBF kernels were used in the first study involving ovarian cancer. In the second stage, two DBN models based G1DBN and LASSO are applied on the best selected features, relationships inferred, comparison made between the two methods and further exploration of inferred relationships carried out from published literature.

In the second study involving hypertension, feature selection was carried out using RFRFE, LASSO, and SVMRFE with linear, RBF and polynomial kernels in the first stage and DBN applied in the second stage. The relationships between the features are inferred and further exploration of inferred relationships carried out from published literature. These case studies are discussed in section 5.3 and section 5.4 with their corresponding results.

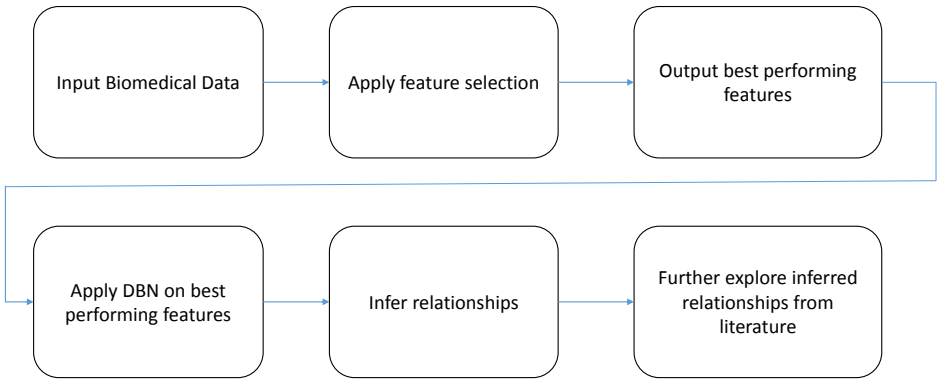

FIGURE 5.1: Block diagram of Computational Model representing the Two-Stage Bio-Network Discovery Approach adapted in this chapter

5.2 Materials and Methods

In this section, a proposed two-stage bio-network discovery approach is applied in two different case studies to prove its efficiency and robustness. The methods used are described thus; at the first stage, different feature selection methods are applied and the best performing features based on statistical significance are selected. At the second stage, dynamic Bayesian network is used to infer the temporal relationships of the best performing features.

5.2.1 Feature Selection using Random Forest Recursive Feature Elimination

Random Forests are ensemble methods for feature classification where many trees are generated using the idea of bagging proposed by [30]. Random Forest computes the ensemble of trees

$$f(x) = \sum_{m=1}^{M} \frac{1}{M} fm(x) \tag{5.1}$$

where f_m is the mth tree and M represents different trees to be trained on different subsets of the data. The feature best among a set of randomly selected features is the feature to be split in each at each node. After a large number of trees have been generated, the features with the most popular class are selected and recursive elimination of least important features is carried out based on decreased classification accuracy.

5.2.2 Feature Selection using Least Absolute Shrinkage and Selection Operator (LASSO)

The lasso developed by Robert Tibshirani in 1996 [98] is a method of regression analysis that performs both regularization and variable selection. It improves prediction performance and prevents over-fitting and for both linear and logistic regression and performs both shrinkage and continuous subset selection via an L_1-norm regularisation penalty. In the R programming environment, the LASSO is implemented by the LiblineaR package [179].

5.3 Bio-Network Discovery Approach for Ovarian Cancer Metabolites

Different computational approaches have been used for ovarian cancer research such as Support Vector Machine [163] and Artificial Neural Network [176]. These methods only account for performance of selected features but do not consider relationships among the features. Also, most biomarker discovery efforts for ovarian cancer focus mainly on large biopolymers such as DNA and RNA but small metabolomic features (below 1KDa) have received little research attention.

The aim of this study was to investigate temporal associations of ovarian cancer features and discover potential time-dependent associations among feature subsets. The experimental hypothesis is that the two-stage bio-network discovery approach involving feature selection and DBN modelling yields better biologically interpretable results with significant relationships than single stage. This means that at the first stage, best performing features are selected and at the second stage, two DBN inference methods will predict significant relationships with some relationships predicted in common between the two DBN algorithms. To test the hypothesis, four different

feature selection methods are used at the first stage to select be performing features. These methods are Support Vector Machine Recursive Feature Elimination with linear kernel (SVMRFE-Linear), Support Vector Machine Recursive Feature Elimination with radial kernel (SVMRFE-Radial), Least Absolute Shrinkage and Selection Operator (LASSO) and Random Forest Recursive Feature Elimination (RFRFE). Two DBN algorithms are used at the second stage which are the LASSO DBN and the G1DBN algorithms.

5.3.1 Results

The ovarian cancer dataset was obtained from [19] and consists of serum samples from 37 papillary serous ovarian cancer patients and 35 controls. Details about the dataset and its description can be found in section 4.2. To determine the important features to be included in the model, feature selection was carried out using four different methods which are: Support Vector Machine Recursive Feature Elimination with linear kernel (SVMRFE-Linear), Support Vector Machine Recursive Feature Elimination with radial kernel (SVMRFE-Radial), Least Absolute Shrinkage and Selection Operator (LASSO) and Random Forest Recursive Feature Elimination (RFRFE). For all four algorithms, 12-fold cross validation was used. Performance criteria used to evaluate the features include sensitivity, accuracy, specificity, type 1 error or the false positive rate (FPR), and Matthews Correlation Coefficient (MCC).

$$Accuracy = (TP+TN)/(TP+TN+FP+FN) \tag{5.2}$$

$$Sensitivity(recall) = TP/(TP+FN) \tag{5.3}$$

$$Precision = TP/(TP+FP) \tag{5.4}$$

$$Specificity = TN/(TN+TP) \tag{5.5}$$

$$Type1error(FPR) = FP/(FP+TN) \tag{5.6}$$

$$Matthews Correlation Coefficient\,(MCC) = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP+FP) \times (TP+FN) \times (TN+FP) \times (TN+FN)}} \tag{5.7}$$

where TP, TN, FP and FN are the true positive, true negative, false positive and false negative respectively.

The NaN values in the cross-validation tables stand for 'Not a Number'. This is how the R programming language denotes occurrences in the computation of the performance criteria where both the numerator and the denominator are both zeros. The MCC is used as a measure of quality in binary classification that takes into account true and false positive and negatives. It is the correlation coefficient between the observed and predicted binary classification and returns a value between -1 and +1. A coefficient of +1 indicates perfect prediction and -1 indicates total disagreement between prediction and observation.

Table 5.1 shows 12-fold cross-validation results of the 40 features selected by RFRFE. The second fold had NaN values in both the Sensitivity and MCC values indicating areas where the numerator and denominator of the equations 5.3 and 5.6 were both zero. Worst performance was at the 11th fold where the accuracy was only 0.3333. This was also reflected by the negative value of the MCC at that fold at -0.2500.

TABLE 5.1: 12-fold Cross-Validation Result of the 40 features selected by RFRFE

Fold	Sensitivity	Specificity	Accuracy	Type 1 error	MCC
1	0.5000	1.0000	0.6667	0.0000	0.5000
2	NaN	0.6667	0.6667	0.3333	NaN
3	0.5000	1.0000	0.8333	0.0000	0.6325
4	0.6667	0.3333	0.5000	0.6667	0.0000
5	0.5000	0.5000	0.5000	0.5000	0.0000
6	1.0000	0.7500	0.8333	0.2500	0.7071
7	0.7500	0.5000	0.6667	0.5000	0.2500
8	1.0000	0.3333	0.6667	0.6667	0.4472
9	0.6667	0.6667	0.6667	0.3333	0.3333
10	0.7500	0.0000	0.5000	1.0000	-0.3162
11	0.2500	0.5000	0.3333	0.5000	-0.2500
12	0.5000	0.5000	0.5000	0.5000	0.0000

Table 5.2 shows the 12-fold cross-validation Result of the 39 features selected by LASSO. The algorithm performed well across multiple folds. Accuracy results across folds 2 to 5, 7 and 10 to 12 were all 1.0000 or 100%. MCC values of 1 across them also meant perfect prediction however results from other folds compensated for over-fitting —one of the main advantages of using cross-validation. It also achieved high sensitivity values of 1.0000 in 11 out of the 12 folds.

Table 5.3 and Table 5.4 show 12-Fold cross-validation results of the top 50 features selected by the SVMRFE-Linear and SVMRFE-Radial. The first two folds of the SVMRFE-Linear showing perfect predictions with sensitivity, specificity, accuracy and MCC values at 1.0000 or 100%. The performance of the SVMRFE-Radial was good with high accuracy value of 1 in the 6th and 7th fold. After the feature selection using the various algorithms, 39 features were selected by the LASSO because they had non-zero coefficients as other less important features are penalised to zero by the LASSO algorithm.

Table 5.5 shows overall performances of all the feature selection algorithms. It showed that the 39 features selected by the LASSO outperformed the others with an overall accuracy of 0.9306. Appendix A.1 shows 8 of these 39 features. For the RFRFE, 40 best features were selected when 2000 trees were initialized for 1000 iterations. For SVMRFE-Linear and SVMRFE-Radial, the performance of top 50 ranked features were considered which had overall accuracies of 0.9167 and 0.6944 respectively.

The relationships of these best performing features were modelled using two dynamic Bayesian Network methods and the resulting networks compared. The first method, the G1DBN [1], [131] is based on vector autoregression while the second method is based on LASSO [180]. The resulting G1DBN network showed the discovery of 65 directed arcs which describe full order conditional dependencies. The range of the score matrices was between 0.000 and 0.049 where lower values represent greater strength and probability in the connection between the edges. Figure 5.2 shows DBN model

TABLE 5.2: 12-fold Cross-Validation Result of the 39 features selected by LASSO

Fold	Sensitivity	Specificity	Accuracy	Type 1 error	MCC
1	1.0000	0.0000	0.6667	1.0000	NaN
2	1.0000	1.0000	1.0000	0.0000	1.0000
3	1.0000	1.0000	1.0000	0.0000	1.0000
4	1.0000	1.0000	1.0000	0.0000	1.0000
5	1.0000	1.0000	1.0000	0.0000	1.0000
6	0.5000	1.0000	0.8333	0.0000	0.6325
7	1.0000	1.0000	1.0000	0.0000	1.0000
8	1.0000	0.5000	0.8333	0.5000	0.6325
9	1.0000	0.5000	0.8333	0.5000	0.6325
10	1.0000	1.0000	1.0000	0.0000	1.0000
11	1.0000	1.0000	1.0000	0.0000	1.0000
12	1.0000	1.0000	1.0000	0.0000	1.0000

TABLE 5.3: 12-Fold Cross-Validation Result of the 50 features selected by SVMRFE-Linear

Fold	Sensitivity	Specificity	Accuracy	Type 1 error	MCC
1	1.0000	1.0000	1.0000	0.0000	1.0000
2	1.0000	1.0000	1.0000	0.0000	1.0000
3	0.6667	1.0000	0.8333	0.0000	0.7071
4	0.8000	1.0000	0.8333	0.0000	0.6325
5	1.0000	1.0000	1.0000	0.0000	1.0000
6	1.0000	1.0000	1.0000	0.0000	1.0000
7	1.0000	1.0000	1.0000	0.0000	1.0000
8	1.0000	0.7500	0.8333	0.2500	0.7071
9	0.6667	1.0000	0.8333	0.0000	0.7071
10	1.0000	0.6000	0.6667	0.4000	0.4472
11	1.0000	0.7500	0.8333	0.2500	0.7071
12	1.0000	1.0000	1.0000	0.0000	1.0000

TABLE 5.4: 12-Fold Cross-Validation Result of the 50 features selected by SVMRFE-Radial

Fold	Sensitivity	Specificity	Accuracy	Type 1 error	MCC
1	0.5000	1.0000	0.6667	0.0000	0.5000
2	0.5000	1.0000	0.8333	0.0000	0.6325
3	0.8000	1.0000	0.8333	0.0000	0.6325
4	1.0000	0.3333	0.6667	0.6667	0.4472
5	1.0000	0.5000	0.8333	0.5000	0.6325
6	1.0000	1.0000	1.0000	0.0000	1.0000
7	NaN	1.0000	1.0000	0.0000	NaN
8	0.8000	1.0000	0.8333	0.0000	0.6325
9	NaN	0.6667	0.6667	0.3333	NaN
10	0.7500	0.5000	0.6667	0.5000	0.2500
11	0.5000	1.0000	0.6667	0.0000	0.5000
12	1.0000	0.0000	0.8333	1.0000	NaN

TABLE 5.5: Overall Performance of Selection Algorithms on Ovarian Cancer Metabolites Dataset

	Performance Measures					
Feature Selection	Classifier	Sensitivity	Specificity	Accuracy	Type 1 error	MCC
RFRFE	Decision Tree	0.7083	0.5375	0.5556	0.4625	0.2163
LASSO	**L1 Logistic Regression**	**0.8788**	**0.9861**	**0.9306**	**0.0139**	**0.8867**
SVMRFE-Linear	SVM Linear	0.8750	0.9079	0.9167	0.0903	0.8558
SVMRFE Radial	SVM Radial	0.7417	0.6758	0.6944	0.3242	0.4212

of the Ovarian cancer metabolite showing the significant features and their temporal relationships. A feedback loop between features 219 and 51 implies that the features influence each other in time and may be of significant clinical importance. The choice of colours in all the diagrams in this chapter are for distinction and clarity.

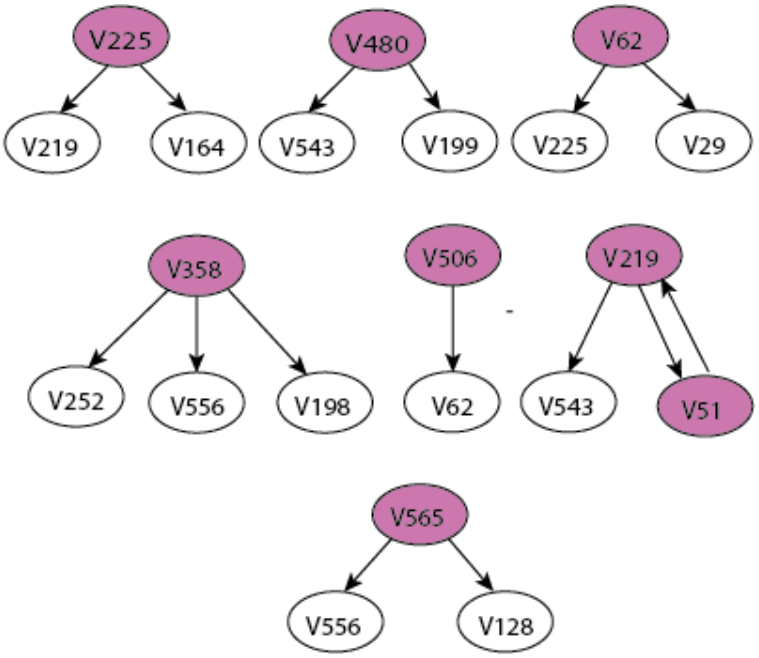

FIGURE 5.2: The Dynamic Bayesian Network Model of key Ovarian Cancer Metabolite features showing time-course relationships across two time points. *(It is important to note that the purple spheres are features of the predicted parents at time t-1 which inhibit the children shown in the white spheres at time t)* [3]

The experiment was performed using the LASSO DBN algorithm. The inferred network was compared with the network generated by the G1DBN model. The result (shown in Figure 5.3) indicate that features 219 and 225, and features 219 and 543, had consistent temporal relationships as inferred by the two algorithms. The mass/charge (m/z) values of features 219, 225 and 543 were found from the original dataset to be 478.3359, 443.3006 and 583.2555 respectively. The metabolites corresponding to these values were found from the Metlin metabolite database [181]. Feature 225 is parent to features 219 and 164 while feature 480 is parent to both feature 543 and 199. Feature 62 is parent to features 225 and 29. The figure also shows that features 252, 556 and 198 have feature 358 as their parent and feature 62 has only feature 506 as its parent. These inferred relationships may be of significant clinical importance and the

consistent arcs inferred by two DBN algorithms should further be investigated in wet laboratories.

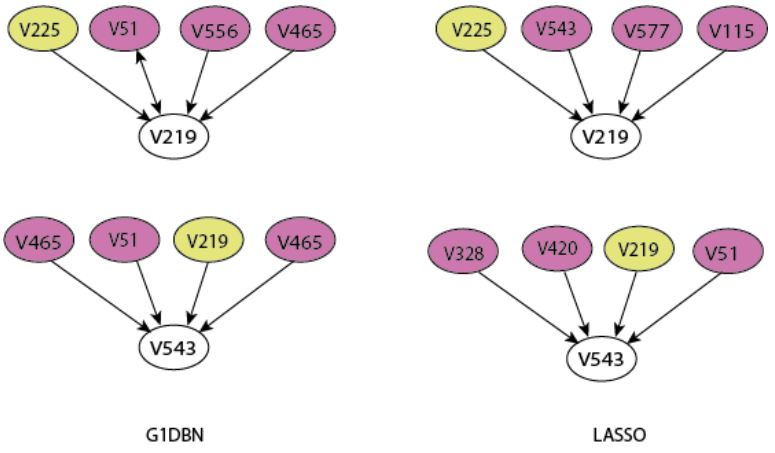

FIGURE 5.3: Comparison of metabolic profiles of ovarian cancer obtained through DBN models based on G1DBN and LASSO algorithms. *(It is important to note that the purple spheres are features of the predicted parents at time t-1 which inhibit the children shown in the white spheres at time t and the yellow spheres are the common features features in the two DBN prediction results)*

Table 5.6 shows description of features 219, 225 and 543 and the corresponding metabolites from the database. It shows that 309 metabolites correspond to feature 219, 733 metabolites correspond to feature 225 and 300 metabolites correspond to feature 543. Both features 219 and 225 have negatively charged adduct ion while feature 543 has positively charged adduct ion.

TABLE 5.6: Description of Key Ovarian Cancer Features

Feature Subsets	**Mode**	**Adduct ion**	**m/z(Da)**	**Number of Metabolites**
V219	Negative	$[M-H]^-$	478.3359	309
V225	Negative	$[M-H]^-$	443.3006	733
V543	Positive	$[M+H]^+$	583.2555	300

Table 5.7 describes some of the metabolites corresponding to feature 543. It further shows two main isomers $C_28H_34N_6O_8$ and $C_28H_38N_6O_8$ with 60 repeats while $C_29H_42N_8O_5$ had 48 repeats. The corresponding change in parts per million of these repeats are 7, 74 and 136 respectively. Stratified analysis showed the isomers were amino acid sequences that potentially make up key ovarian cancer metabolites [181].

TABLE 5.7: Description of Metabolites for Feature 543

Change in ppm	Number of Repeats	Formula
0	1	$C_{33}H_{34}N_4O_6$
7	60	$C_{28}H_{34}N_6O_8$
17	24	$C_{24}H_{38}N_8O_7S$
24	24	$C_{29}H_{38}N_6O_5S$
29	2	$C_{28}H_{38}O_{13}$
38	24	$C_{28}H_{34}N_6O_6S$
39	7	$C_{35}H_{34}O_8$
42	1	$C_{35}H_{38}N_2O_6$
46	1	$C_{30}H_{34}N_2O10$
54	24	$C_{29}H_{38}N_6O_7$
55	1	$C_{26}H_{47}O_{12}P$
61	1	$C_{34}H_{38}N_4O_5$
73	1	$C_{18}H_{38}N_4O_{15}S$
74	60	$C_{28}H_{38}N_8O_6$
86	12	$C_{22}H_{38}N_{12}O_7$
91	1	$C_{27}H_{34}O_{14}$
101	1	$C_{34}H_{30}O_9$
121	1	$C_{24}H_{46}O_8$
127	2	$C_{30}H_{30}O_{12}$
136	48	$C_{29}H_{42}N_8O_5$
153	4	$C_{26}H_{30}O_{15}$

5.4 Hypertension Diagnosis via Temporal inference of Gene Expression Profiles

The proposed two-stage approach was further applied to gene expression profiles of high quality hypertension datasets in order to determine temporal associations among features which may be important in the diagnosis of the disease.

Hypertension or high blood pressure is a common disease with one in three adults in the United States of America affected [182]. It is defined as the diastolic or systolic blood pressure greater than or equal to 90mmhg and 140mmhg respectively. It is also considered to be a silent killer and a major risk factor for developing heart disease and

as many as seven million people in the UK currently are living with undiagnosed high blood pressure [183].

Various computational approached had been applied in trying to effectively diagnose the disease such as improved C4.5 classification algorithm [177] based on the use of maximal information for the identification of important features. Other methods such as fuzzy systems and neuro-fuzzy systems [178] have also been applied in an attempt to better diagnose the disease.

To overcome the inability of these methods to incorporate and model the temporal relationships among key hypertension features, the proposed two-stage bio-network discovery approach was applied. In this study, five feature selection algorithms were applied at the first stage. These are Random Forest Recursive Feature Elimination (RFRFE), Least Absolute Selection and Shrinkage Operator (LASSO), Support Vector Machine Recursive Feature Elimination with Linear Kernel (SVMRFE-Linear), Support Vector Machine Recursive Feature Elimination with Polynomial Kernel (SVMRFE-Poly) and Support Vector Machine Recursive Feature Elimination with Radial Kernel (SVMRFE-Radial). At the second stage, dynamic Bayesian network defining a vector autoregression of order 1 VAR(1) was used to infer the relationships.

5.4.1 Results

The dataset was obtained from [20] and comprise of gene expression profile of male young-onset hypertension of ages 20-50 years. Section 4.3 explains the details of the data. To determine which features to model their relationships, feature selection using the five different algorithms was carried out. This ensured features with best class discriminatory information were selected. For all selection algorithms, 10-fold cross validation was performed and average performance measures taken. Performance criteria used include accuracy, sensitivity, specificity, balanced classification rate (BCR),

Type 1 error or False Positive Rate (FPR), F1-score and Matthews Correlation Coefficient (MCC).

$$F1 - Score = 2 \times ((Precision * Recall)/(Precision + Recall)) \tag{5.8}$$

$$\begin{gathered} Balanced\,Classification\,Rate\,(BCR) = \\ 1/2 \times (TP/(TP+FN)) + (TN/(TN+FP)) \end{gathered} \tag{5.9}$$

where F1-Score means the harmonic mean between the precision (the positive predicted value) and the recall (sensitivity) and the Balanced Classification Rate indicates a balanced measure between the values of the sensitivity and the True Negative Rate (the proportion of actual negatives which are predicted negative).

Table 5.8 shows 10-Fold Cross-Validation result of the 320 features selected by RFRFE. It showed highest accuracy value of 0.7333 at folds 2, 5 and 9. Fold 1 had the lowest BCR of 0.4722, low specificity of 0.1667 and a negative MCC value of 0.0680. Table 5.9, Table 5.10 and Table 5.11 show the performance of the SVM across the linear, polynomial and radial basis function kernels respectively. It indicates very good performance by the linear kernel which outperformed both the radial and the polynomial kernels. The performance of the linear kernel showed perfect predictions across folds 1 to 3 and 5 to 9. However, the performance of the 101 features selected by LASSO because they are the non-zero coefficients exceeded the good performance of the SVMRFE with linear kernel. It showed perfect performance in 9 out of the 10 folds and had overall best result.

Table 5.13 shows the number of selected features by the algorithms and their performance measures. For the SVMRFE methods, the cost of constraint parameter C was varied between between 1 and 10 and $C = 1$ with best accuracy was used. Moderate polynomial of degree 3 was used for the SVMRFE-Poly with a gamma value

TABLE 5.8: 10-Fold Cross-Validation result of the 320 features selected by RFRFE

Fold	Sensitivity	Specificity	Accuracy	Type 1 error	BCR	F1-Score	MCC
1	0.7778	0.1667	0.5333	0.8333	0.4722	0.6667	-0.0680
2	0.6250	0.8571	0.7333	0.1429	0.7411	0.7143	0.4910
3	0.8571	0.5000	0.6667	0.5000	0.6786	0.7059	0.3780
4	0.5000	0.6364	0.6000	0.3636	0.5682	0.4000	0.1231
5	0.6667	0.8333	0.7333	0.1667	0.7500	0.7500	0.4910
6	0.4444	0.6667	0.5333	0.3333	0.5556	0.5333	0.1111
7	0.8000	0.5000	0.6000	0.5000	0.6500	0.5714	0.2887
8	0.7143	0.5000	0.6000	0.5000	0.6071	0.6250	0.2182
9	0.6250	0.8571	0.7333	0.1429	0.7411	0.7143	0.4910
10	0.4615	0.5000	0.4667	0.5000	0.4808	0.6000	-0.0262

TABLE 5.9: 10-Fold Cross-Validation result of the 137 features selected by SVMRFE Linear

Fold	Sensitivity	Specificity	Accuracy	Type 1 error	BCR	F1-Score	MCC
1	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
2	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
3	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
4	1.0000	0.8889	0.9333	0.1111	0.9444	0.9231	0.8729
5	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
6	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
7	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
8	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
9	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
10	0.9000	1.0000	0.9333	0.0000	0.9500	0.9474	0.8660

TABLE 5.10: 10-Fold Cross-Validation result of the 45 features selected by SVMRFE-Poly

Fold	Sensitivity	Specificity	Accuracy	Type 1 error	BCR	F1-Score	MCC
1	1.0000	0.8889	0.9333	0.1111	0.9444	0.9231	0.8729
2	0.7273	1.0000	0.8000	0.0000	0.8636	0.8421	0.6447
3	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
4	1.0000	0.9000	0.9333	0.1000	0.9500	0.9091	0.8660
5	0.8889	0.8333	0.8667	0.1667	0.8611	0.8889	0.7222
6	0.8000	0.9000	0.8667	0.1000	0.8500	0.8000	0.7000
7	0.8333	1.0000	0.9333	0.0000	0.9167	0.9091	0.8660
8	1.0000	0.8889	0.9333	0.1111	0.9444	0.9231	0.8729
9	0.6667	0.8333	0.7333	0.1667	0.7500	0.7500	0.4910
10	1.0000	0.7500	0.9333	0.2500	0.8750	0.9565	0.8292

TABLE 5.11: 10-Fold Cross-Validation result of the 49 features selected by SVMRFE Radial

Fold	Sensitivity	Specificity	Accuracy	Type 1 error	BCR	F1-Score	MCC
1	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
2	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
3	1.0000	0.8889	0.9333	0.1111	0.9444	0.9231	0.8729
4	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
5	0.9000	1.0000	0.9333	0.0000	0.9500	0.9474	0.8660
6	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
7	1.0000	0.8571	0.9333	0.1429	0.9286	0.9412	0.8729
8	1.0000	0.8889	0.9333	0.1111	0.9444	0.9231	0.8729
9	1.0000	0.8000	0.9333	0.2000	0.9000	0.9524	0.8528
10	0.8889	1.0000	0.9333	0.0000	0.9444	0.9412	0.8729

TABLE 5.12: 10-Fold Cross-Validation result of feature selection using LASSO

Fold	Sensitivity	Specificity	Accuracy	Type 1 error	BCR	F1-Score	MCC
1	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
2	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
3	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
4	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
5	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
6	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
7	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
8	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
9	1.0000	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000
10	0.9000	1.0000	0.9333	0.0000	0.9500	0.9474	0.8660

TABLE 5.13: Summary of Best Performance Criteria on Hypertension Dataset

Performance Measures									
Feature Selection	Classifier	no. of genes	Sensitivity	Specificity	Accuracy	Std.Acc	FPR	BCR	F1-Score	MCC
RFRFE	Decision Tree	320	0.6292	0.5854	0.6067	0.1215	0.4146	0.6073	0.5965	0.2164
LASSO	**L1 Logistic Regression**	**101**	**0.9587**	**1.0000**	**0.9900**	**0.0211**	**0.0139**	**0.9923**	**0.9929**	**0.9873**
SVMRFE-Linear	SVM-Linear	137	0.9889	0.9850	0.9900	0.0281	0.0145	0.9894	0.9850	0.9739
SVMRFE-Poly	SVM-Poly	45	0.9260	0.8470	0.8867	0.0892	0.1530	0.8865	0.8949	0.7800
SVMRFE-RBF	SVM-RBF	49	0.9975	0.9514	0.9600	0.0562	0.0486	0.9595	0.9531	0.9197

of 0.0000451 which corresponds to the inverse of the dataset's dimension. With the SVMRFE-Linear, the SVMRFE-Poly and SVMRFE Radial, 137, 45 and 49 best features were selected respectively. Using the RFRFE, 1000, 2000 and 5000 forests were randomly initialised and the experiment conducted for 200, 500 and 1000 iterations. 320 top features were selected by the random forest method based on backward elimination. The experiment results showed that 101 features which were selected by the LASSO had best performance followed by the 137 features selected by the SVMRFE Linear. Both had 99% accuracy however the features selected by the LASSO had higher specificity and lower false positive rate than those selected by the SVMRFE Linear.

For modelling the temporal associations of selected genes, the 101 and 137 features selected by the LASSO and the SVMRFE Linear were considered as they both had much higher accuracies of 99% while the other features were disregarded. The DBN inference was performed using the G1DBN algorithm [131] based on first order conditional dependencies and vector autoregression. There are two main steps involved in the inference algorithm. In step 1, a first order dependency score matrix (S1) is inferred which contains the score of each edge of the DBN. The inference is done by defining an edge selection threshold $\alpha 1$. At step 2, edge selection threshold $\alpha 1$ and score matrix S1 are used to infer the score of the edges of a DBN defining full order conditional dependencies between successive variables. The smallest score refers to the most significant edge. To obtain optimised DBN, threshold values of $\alpha 1$ and $\alpha 2$ are found to be 0.5 and 0.05 respectively which are used to prune the edges of the DBN [4].

The result of the inferred network showed the discovery of 351 directed arcs defining full order conditional dependencies for the 101 features selected by the LASSO and 460 directed arcs for the 137 features selected by the SVMRFE Linear.

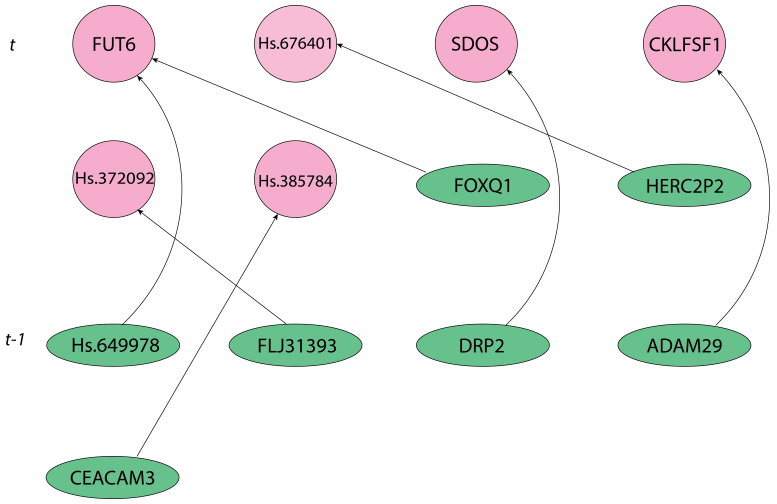

FIGURE 5.4: This shows the DBN model of Hypertension genes showing temporal relationships 13 strongest edges and their connecting genes. It is worth noting that the predicted parents at time *t-1* are shown by the green ellipses. These features inhibit the children shown in the pink circles at time *t* [4]

Figure 5.4 shows the DBN inference of 13 strongest edges and their connecting features selected by the LASSO. These have most significant edges with a score 0.000. The meaning of the genes with symbols and genes represented only by their UniGene IDs (starting with Hs.) were verified from Gene Expression Omnibus [184]. The figure shows that the human transcribed locus gene (Hs.649978) and cell regulation may inhibit the expression of focusyltransferase-6 (FUT6).

Figure 5.5 shows DBN inference of the 15 genes with the most significant inferred edges having scores of 0.0000. It shows that both NR2E3 nuclear receptor sub-family 2 and CGB2 Chromatin Gonadotrophin might play inhibitory role against ABCG1 ATP-binding cassette sub-family G. Further stratified analysis showed that 22 genes were picked up in common by both the LASSO and the SVMRFE linear feature selection methods.

Figure 5.6 shows temporal associations of 11 genes that have six most significant edges of lowest scores. The result showed that CRABP2 cellular retinoic acid binding protein 2 may play an inhibitory role and highly associated with DKK3 dickkopf WNT signaling pathway inhibitor 3.

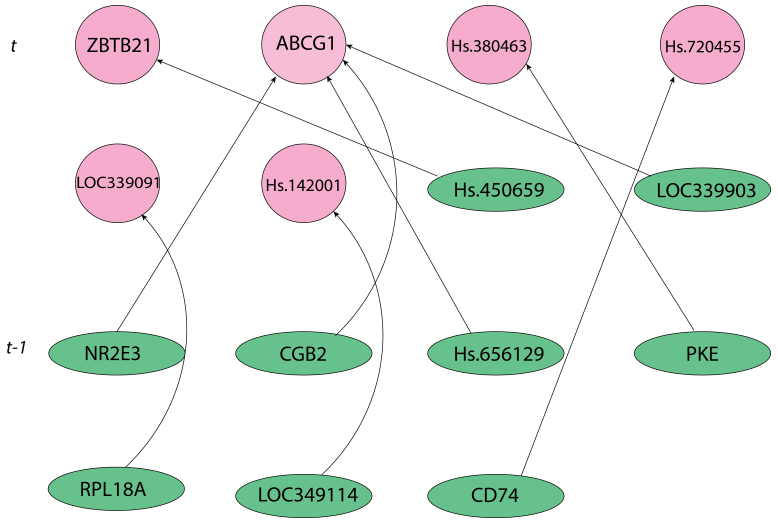

FIGURE 5.5: DBN Model of Hypertension genes showing temporal relationship among top genes selected by SVM-RFE Linear method. *It is worth noting that the predicted parents at time t-1 are shown by the green ellipses. These features inhibit the children shown in the pink circles at time t*

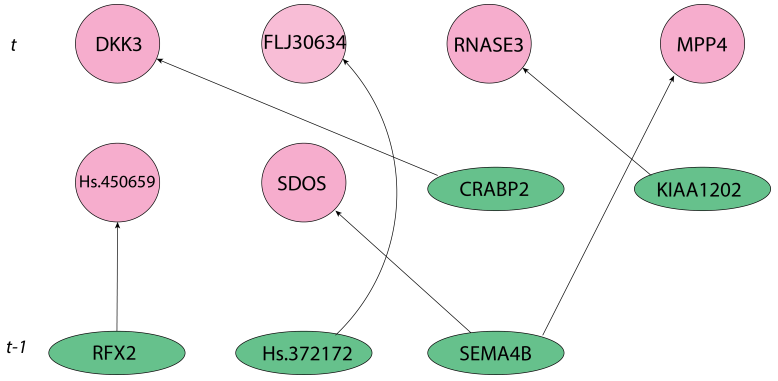

FIGURE 5.6: DBN of Hypertension genes showing temporal relationship among key features selected by both the LASSO and SVM-RFE Linear methods (the green spheres are features of predicted parents at time *t-1* which inhibit features in red circles (children) at time *t*.

Four key genes were selected in common by the three SVMRFE methods and the LASSO as shown in Table 5.14. These are Human transcribed locus with UniGene ID Hs.666652, ribonuclease RNase A family 3 (eosinophil cationic protein) (RNASE3), Human protein-coding gene (PLXNA2) and CDNA FLJ36210 fis, clone THYMU2000155 with UniGene ID Hs.656129. These genes selected in common across different selection methods may be highly associated with hypertension and should further be investigated. Appendix A.2 shows description of the 101 hypertension features selected by the LASSO.

TABLE 5.14: Key Hypertension Genes Commonly Selected by LASSO and SVM-RFE Methods

UniGene ID	Symbol	Name
Hs.666652	null	Human transcribed locus
Hs.73839	RNASE3	ribonuclease; RNase A family; 3 (eosinophil cationic protein)
Hs.497626	PLXNA2	Human protein-coding gene PLXNA2
Hs.656129	null	CDNA FLJ36210 fis, clone THYMU2000155

5.5 Conclusion

In this study, a two-stage bio-network discovery approach was developed and applied in two different real world case studies. The framework allowed for modelling the relationships across high quality selected features using a dynamic Bayesian network as an inference algorithm. The experimental hypothesis is that two-stage bio-network discovery approach involving feature selection and Dynamic Bayesian Network (DBN) modelling yields better biologically interpretable results with significant relationships than single stage. This means that at the first stage, best performing features are selected and at the second stage, DBN inference methods will predict significant relationships. To test the hypothesis, four different feature selection methods are used at the first stage to select be performing features. Two DBN methods are used in the second stage and it was hypothesised that there would be similarities in the results of the two DBN methods with some relationships predicted in common between the two DBN algorithms.

Feature selection was carried out using 12-fold cross-validation. The cross-validation tables show how the features performed across each fold by the different feature selection algorithms. Mean cross-validation results showed that the 39 features selected by the least absolute shrinkage and selection operator (LASSO) had the highest accuracy of 93.06%. The performance of these features was better than other the performance of

the features selected by other algorithms which are the random forest recursive feature elimination and the support vector machine recursive feature elimination with linear and radial basis function kernels. The 39 features were selected by the (LASSO) because they had non-zero coefficients. 7 of these selected features are shown in section 1 of Appendix A.

Their performance were assess using criteria which include accuracy, sensitivity, specificity and Matthews Correlation Coefficient. Feature selection algorithms used include: Support Vector Machine Recursive Feature Elimination with linear kernel, Support Vector Machine Recursive Feature Elimination with radial kernel (SVMRFE-Radial), Least Absolute Shrinkage and Selection Operator (LASSO) and Random Forest Recursive Feature Elimination (RFRFE). 39 features were selected by the LASSO because they had non-zero coefficients as other less important features are penalised to zero by the LASSO algorithm. These had best overall performance when compare to the features selected by other methods.

The relationships of these best performing features were modelled using two dynamic Bayesian Network methods and the resulting networks compared. These relationships could reveal novel pathways in the metastasis of the disease and aid in further diagnosis and drug discovery. The study also shows comprehensive comparison of two DBN inference algorithms, the G1DBN and the LASSO, and how similar result obtained by the two may be helpful *in insilico* modelling of biological pathways.

The second case study used the proposed two-stage bio-network discovery approach to analyse temporal associations of Hypertension gene expression profile dataset. Methods applied include Support Vector Machine Recursive Feature Elimination using the linear, polynomial and radial basis function kernels of the Support Vector Machine, Random Forest Recursive Feature Elimination and Least Absolute Selection and Shrinkage Operator (LASSO).

10-fold cross-validation tables presented show performances of the selected features. The tables indicate very good performance by the linear kernel which outperformed both the radial and the polynomial kernels. The performance of the linear kernel showed perfect predictions across folds 1 to 3 and 5 to 9. However, the performance of the 101 features selected by LASSO because they are the non-zero coefficients exceeded the good performance of the SVMRFE with linear kernel. It showed perfect performance in 9 out of the 10 folds and had overall best result.

The findings of this chapter appear to suggest that further investigation into the features which showed self loops, and those whose relationships showed the same results using the two DBN modelling methods are required. One major challenge of the two-stage approach approach explored in this chapter seems to be robust and automatic selection of best set of parameters for the feature selection algorithms to be applied at the first stage instead of performing multiple runs of the experiment using different manually selected parameter values. Therefore, chapter 6 aims to address this problem.

Chapter 6

Improved prediction of Gene Regulatory Networks via Optimised Two-Stage Approach for Cancer Diagnosis

6.1 Introduction

In chapter 5, a two-stage bio-network discovery approach was proposed and developed. The efficiency of the proposed method was tested in two different real world case studies. However, parameter values for the feature selection algorithms at the first stage were manually fixed which required many runs of the experiment to achieve good results based on values that yielded best accuracy.

This chapter introduces the development of a hybrid approach by adapting parameter optimisation algorithms. Instead of trying out every possible combination of parameters using popular grid search method, combination of parameters that yield best

results are evolved. This is more efficient and saves time when a wide range of parameter combinations are needed. This was successfully implemented for the study of colorectal cancer metastasis and results showed improved performance when compared with non-optimised approaches. The hypothesis is that optimised embedded methods of feature selection are better than non-optimised filter methods of feature selection in colorectal cancer diagnosis. Filter methods are selection methods that ignore interaction with the classification algorithm. Embedded methods are selection methods that interact with classification algorithm. This chapter also investigates biological interpretation of best colorectal protein features using two-stage bio-network discovery approach developed in chapter 5. Introduction and motivation for the experiments done in this chapter are described thus.

Colorectal cancer is a deadly disease and one of the most common cancer diagnosed in the UK. It is also known as bowel cancer or rectal cancer depending on where it starts. With around 40,000 new cases [185], and is estimated to cause 49,190 deaths in 2016 [186]. The diagnosis of the disease was traditionally being done using microscopic analysis of histopathological cancer samples. This manual process has necessitated the need for automatic detection and diagnosis using computational means. There are five main detection methods studied [187]. These are object-oriented texture analysis, serum analysis, texture analysis, gene analysis and spectral analysis. Serum analysis in line with recent computational advances in the post-genome era, has received significant attention.

Various methods for computationally diagnosis the disease were proposed. Hong et al 2011 [188] proposed the method based on empirical mode decomposition (EMD) with least square support vector machine (LS-SVM). This involved using EMD for feature selection and LS-SVM for classification of selected features and computation of performance measures using accuracy metric. A feed-forward neural network model was proposed by [189] involved the use of two feature selection methods one after another. The selection methods used are chi-squared method and minimum redundancy

maximum relevance (MRMR) method. A method based on discrete wavelength transformation was also proposed by [169] in which features are selected as wavelength coefficients and classification carried out using support vector machine.

These methods have a number of inadequacies. First, they implement filter method of feature selection which ignore interaction with the classifier and has been known to result in poorer features being selected [190]. Secondly, parameter optimization for feature selection or classification algorithms was not implemented which may have improved overall efficiency and accuracy. Finally, the methods used do not consider interaction between features. This is important as various stages of cancer growth and metastasis are linked to the various stages of molecular interactions of key features. Capturing these interactions using the best optimised way is therefore important in the diagnosis of the disease and further discovery of significant pathways that may be of clinical importance.

This study aims to address these aforementioned problems using an improved two-stage bio-network discovery approach via parameter optimisation. Two major evolutionary algorithms are adopted in this study. These are the Differential Evolution (DE) and Particle Swarm Optimisation (PSO) algorithms. This study extends previous optimised two-stage studies in which only DE was applied using for liver cancer diagnosis [16] and only PSO was applied for colorectal cancer diagnosis [191].

Figure 6.1 shows overall block diagram of the computational model adapted in this chapter. It represents the optimised approach implemented specifically on the colon cancer dataset which describes steps taken to generate and analyse the results. At the first stage it takes cancer data as input and the optimization algorithm generates initial population of optimisation parameters C and gamma for the SVM. The parameters are varied by the optimisation algorithm until lowest error where the error is the objective function to be minimised. Within the loop, recursive feature elimination is carried out to select best features using best parameters. Features with smallest ranking criterion,

in this case the weight magnitude of the SVM, are identified, removed and the best performing features returned as output of the first stage.

At the second stage DBN is then applied on the selected best features, relationships inferred and further exploration of inferred relationships carried out from published literature. The rest of the chapter describes the details of how this model was adapted with corresponding results. It is worth noting that the outcome of this chapter appeared in IET Systems Biology Journal in December 2015 [15].

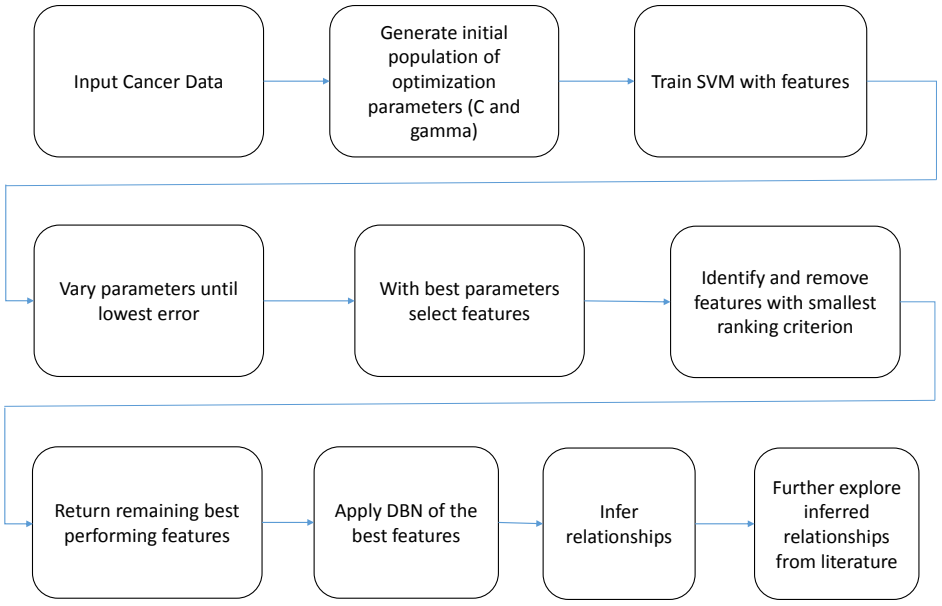

FIGURE 6.1: Block diagram showing Computational Model of the Optimized Two-Stage Approach adapted in this chapter

6.2 Materials and Methods

The colorectal cancer dataset was first presented in [21]. 66 colorectal cancer patients and 50 healthy volunteers were involved in the study whose serum samples were obtained a day before surgery. Further low level analysis which include recalibration, top-hat base line correction and outlier detection were performed on the dataset [169]. Detailed description of the dataset are in section 4.4. The aim of the experiment is

to select best features relevant to the disease and model relationships among features using dynamic Bayesian network via the proposed two-stage approach.

6.2.0.1 Application of the Proposed two-stage approach

In this study, the proposed two stage approach is improved upon through parameter optimisation. It combines optimised SVMRFE feature elimination whose parameters have been fine-tuned using two evolutionary algorithms namely particle swarm optimisation (PSO) and differential evolution (DE). Evolutionary algorithms have been chosen and preferred for the feature selection over grid search due to their exhaustive search for candidate solutions and better overall performance [192] [193]. This will aid the selection of high quality potential biomarkers and also make for better understanding the tumour metastasis which may provide new insight for discovery of new drugs to clinicians and wet-lab scientists.

6.2.0.2 First Stage —Parameter Optimisation for SVMRFE

In the first stage, the C and γ parameters of SVMRFE feature selection algorithm are fine-tuned for best selection performance using the Average Test Error (ATE). The ATE is defined by

$$Accuracy = (TP+TN)/(TP+TN+FP+FN) \tag{6.1}$$

$$ATE = 1 - Accuracy \tag{6.2}$$

where TP, TN, FP and FN are the true positive, true negative, false positive and false negative respectively. The SVMRFE algorithm uses backward elimination based on the weight magnitude of the classifier and performs more accurate selection because

the algorithm interacts with the classifier [28]. This embedded method of feature selection is preferred over the filter method used in previous studies because they have been known to yield better selection results [190]. The SVMRFE procedure involves

- Training the classifier (i.e. optimising the weights w_i for all features.
- Calculating the ranking criterion.
- Removing the features with the smallest ranking criterion.

Cross-validation was performed using the Leave-one-out cross-validation (LOOCV). The SVMRFE uses the optimised parameters for accurate selection of the features. Algorithm 6.1 describes the optimisation approach used in this study.

Algorithm 6.1: SVMRFE Optimisation Algorithm

---

1 Rank features with SVMRFE.

2 Split data with two-thirds for training and one third for testing.

3 Generate an initial population of SVM parameters using PSO and DE Algorithms.

4 With training data, evaluate the ATE as the optimisation objective of each solution using LOOCV.

5 Obtain generalisation error at each generation.

6 Continue until the lowest possible error or no improvement is achieved.

7 Choose parameters with best average performance.

8 Obtain generalisation performance on the test set using optimised parameters.

9 Choose optimum number of features based on number of features with lowest test error.

The algorithm is a slightly modified version of earlier proposed methods [194], [195] due to the implementation of population-based algorithms which are used on the training data to fine-tune the parameters of the SVMRFE. Feature selection using the optimised parameters is then carried out using LOOCV. The LOOCV was chosen over the k-fold cross-validation because of the small sample size of 112.

6.2.0.3 Results from the first Stage

Three main kinds of SVM kernels, the linear, the polynomial and the radial basis function (RBF), were considered for feature selection using recursive feature elimination. The five DE algorithm variants and the PSO algorithm were used and implementation was done using the R Language for Statistical Computing [17]. The aim was minimisation of the ATE objective function. Each algorithm was executed for 40,000 fitness evaluations over 30 independent runs. Manually tuned parameter of the DE were: scaling factor F=0.8 and crossover rate CR=0.9. For the PSO, c1=0.4, c2=0.8, w_max=1.0, w_min=0.1.

The optimisation algorithms were applied on the three different SVMRFE methods. Table 6.1 shows best cross-validation performance results and the optimum values of SVM C and γ parameters. It shows that the PSO performance with the linear kernel of the SVM had the lowest cross validation error of 0.038 followed by the performance of the DE/rand/1 algorithm using the RBF kernel. The DE/rand-to-best/1 had the worst performance on average when compared with the other algorithms. Its performance on the polynomial and RBF kernels were 0.378 and 0.392 respectively with the performance on the linear kernel being 0.486. DE/best/1 had good performance on RBF kernel at 0.169 which was slightly better then the performance of DE/best/2 on the RBF kernel at 0.189. Though the PSO achieved better results than the DE and its variants in this study, the run time of DE was faster than the PSO however, run time is only significant if the resulting solution is of good quality. In this study, the PSO

achieved better and more accurate result than the DE. For this reason the PSO and another variant of PSO called the Comprehensive Learning Particle Swarm Optimisation (CLPSO) were subsequently used in chapter 7 of this thesis.

The variation in performance results of the various SVM kernels was due to the different values of C and γ parameters that were selected after 40,000 fitness evaluations. The γ parameter defines how far the influence of a single training example reaches the decision boundary. Low values of γ means a far reach while higher values of γ indicate a close reach to the decision boundary. The C cost of constraint parameter controls the trade-off between smooth decision boundary and classifying training points correctly. Hence for both linear and nonlinear kernels, the optimum values reached at the end of each iteration of the optimisation algorithms determine the final performance of the SVM kernel.

Table 6.2 shows the values of the optimised parameters. From the table, different algorithms performed differently on the kernels. The PSO and DE/rand/1 achieved lowest test error on the SVM linear and SVM radial kernels respectively. The performance of the DE/rand/2 on the SVM radial kernel closely matches the previous result with a low test error of 0.036. The DE/rand-to-best/1 has overall worst performance with the performance on the linear kernel at 0.243 while the polynomial and RBF kernels were 0.162 and 0.108 respectively.

The area under the curve (AUC) of the receiver operating characteristics (ROC) gives a clearer picture of the performances of the classifiers where a higher value of AUC will indicate better performance of the classifier. The accuracy metric favours the dominant class in feature selection tasks. This means that accuracy values may not always vary with the values of the AUC and high values of the AUC such as 1 (or 100%) does not necessarily indicate absence of misclassification errors which may still persist in the computation depending on the dominant class. The curves of the ROC the values of

AUC show trade-offs between false positive rate and true positive rate values which has been shown to be a better measure than accuracy [196].

From Table 6.2, the AUC value of the linear SVM as optimised by the PSO has a value of 1. This means that it has more probability of better classification performance than the other SVM classifiers. This may have been due to the high-dimension of the dataset and small sample size. The radial kernel however had values of 0.993 when parameters were optimised by DE/rand/2 and 0.988 when optimised by DE/rand/1, DE/best/1 and DE/best/2 respectively.

TABLE 6.1: Cross- Validation results showing best performance with lowest error and best SVM parameters

Optimization Algorithm	SVM-kernel	C	γ	Cross-Validation Error
PSO	Linear	3.0924	—	**0.038**
	Poly	281.334	0.2321	0.319
	RBF	635.783	0.0002	0.125
DE/rand/1	Linear	8.6754	—	0.262
	Poly	75.6499	0.1632	0.165
	RBF	332.3389	0.0003	**0.042**
DE/best/1	Linear	67.9443	—	0.219
	Poly	175.6499	0.1642	0.349
	RBF	7992.5412	0.1566	0.169
DE/rand-to-best/1	Linear	463.9081	—	0.486
	Poly	862.0241	0.3321	0.378
	RBF	4.9882	0.0451	0.392
DE/best/2	Linear	51.5332	—	0.295
	Poly	966.9021	0.0034	0.267
	RBF	1214.9883	0.0043	0.189
DE/rand/2	Linear	20.6754	—	0.196
	Poly	289.8233	0.0009	0.298
	RBF	1279.3341	0.0022	**0.049**

The ROC curves indicating the performances of the optimisation algorithms on the SVM methods is shown in Figure 6.2. It showed the varying performances of the algorithms on the SVM kernels and on the top left, the linear SVM as optimised by the PSO had the highest ROC value of 1. The number of features selected by the

SVMRFE methods using the optimised parameter values are shown in Table 6.3. The PSO optimised SVMRFE linear and the DE/rand/1 optimised SVMRFE radial both had 18 and 25 feature selected respectively. They both achieved high accuracies of 0.973 each and sensitivity values of 1 and 0.952 respectively. 12 of the 18 Colorectal Cancer Spectral Profiles selected by PSO SVMRFE Linear are show in Appendix A.3. There were also 62 features selected by the SVMRFE radial kernel as optimised by DE/rand/2 which achieved a total accuracy of 0.964 and a sensitivity of 1. Other performance values computed are balanced classification rate (BCR), F1-score, specificity and Matthew's Correlation Coefficient (MCC). The corresponding equations are shown:

$$F1 - Score = 2 \times ((Precision * Recall)/(Precision + Recall)) \tag{6.3}$$

TABLE 6.2: Optimised parameters of the PSO and DE algorithms for the three kernels of the SVMRFE showing values of ATE, accuracy and AUC

Optimization Algorithm	SVM-kernel	C	γ	Testing Error	Acc.	AUC
PSO	Linear	**3.0924**	—	**0.027**	0.973	1.000
	Poly	281.334	0.2321	0.162	0.878	0.935
	RBF	635.783	0.0002	0.081	0.919	0.935
DE/rand/1	Linear	8.6754	—	0.189	0.811	0.903
	Poly	75.6499	0.1632	0.054	0.946	0.93
	RBF	**332.3389**	**0.0003**	**0.027**	**0.973**	**0.988**
DE/best/1	Linear	67.9443	—	0.135	0.865	0.962
	Poly	175.6499	0.1642	0.162	0.838	0.938
	RBF	7992.5412	0.1566	0.081	0.919	**0.988**
DE/rand-to-best/1	Linear	463.9081	—	0.243	0.957	0.835
	Poly	862.0241	0.3321	0.162	0.838	0.955
	RBF	4.9882	0.0451	0.108	0.892	0.962
DE/best/2	Linear	51.5332	—	0.108	0.892	0.942
	Poly	966.9021	0.0034	0.054	0.946	0.93
	RBF	1214.9883	0.0043	0.054	0.946	**0.988**
DE/rand/2	Linear	20.6754	—	0.054	0.946	0.971
	Poly	289.8233	0.0009	0.162	0.838	0.9
	RBF	**1279.3341**	**0.0022**	**0.036**	**0.964**	**0.993**

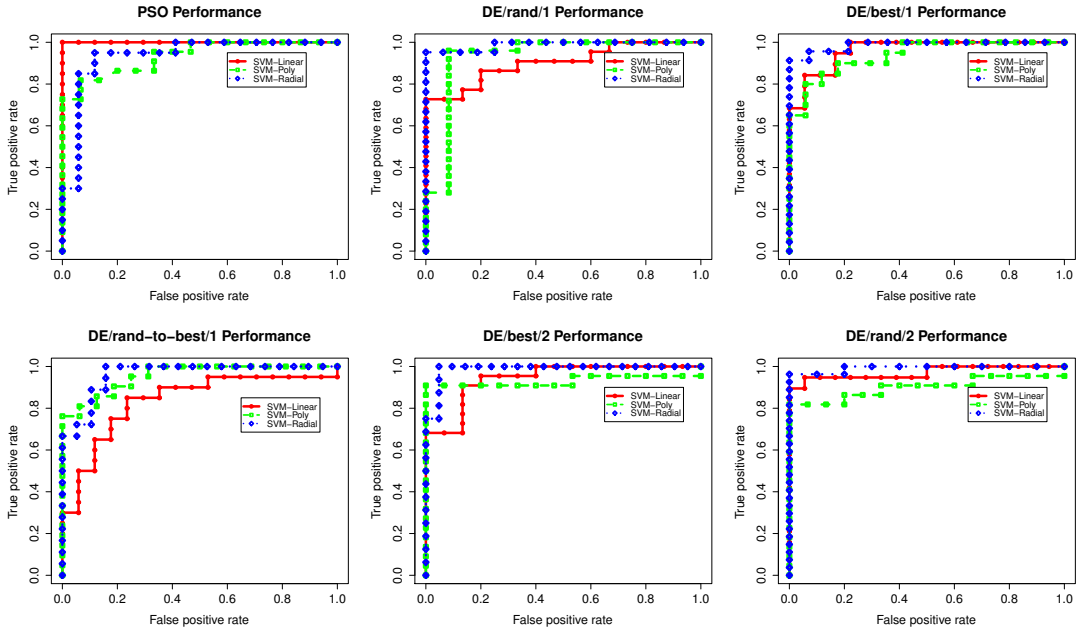

FIGURE 6.2: ROC curves showing the performance of the SVM kernels as optimised by the six optimisation algorithms.

$$Precision = TP/(TP+FP) \tag{6.4}$$

$$Recall(Sensitivity) = TP/(TP+FN) \tag{6.5}$$

$$Specificity = TN/(TN+TP) \tag{6.6}$$

$$Balanced\,Classification\,Rate\,(BCR) = 1/2 \times (TP/(TP+FN)) + (TN/(TN+FP)) \tag{6.7}$$

$$Matthews\,Correlation\,Coefficient\,(MCC) = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP+FP) \times (TP+FN) \times (TN+FP) \times (TN+FN)}} \tag{6.8}$$

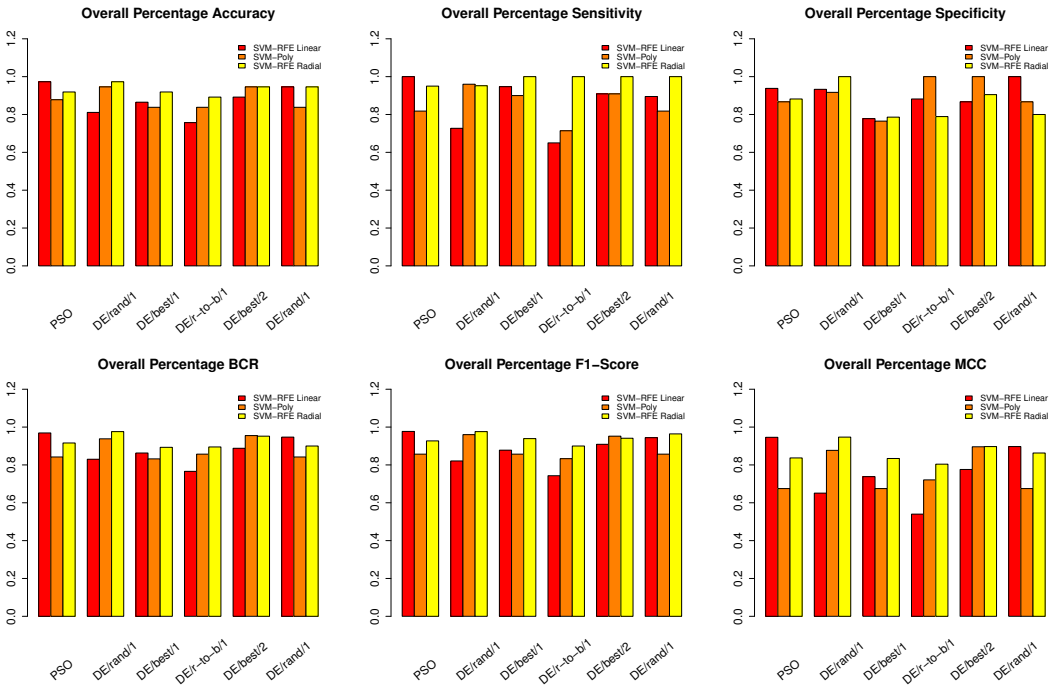

FIGURE 6.3: Overall performance of other performance measures for all SVM methods with all the optimisation algorithms. DE/r-to-b/1 is the DE/rand-to-best/1 algorithm.

where TP is the True Positive, FP is the False Positive, TN is the True Negative and FN is the False Negative. The F1-score is the harmonic mean of the precision and recall. The recall (sensitivity) is the number of true positives divided by the sum of true positive and false negatives and the precision or positive predicted value is the number of true positives divided by the sum of the true positives and false positives [142]

The 18 features selected by the PSO optimised Linear SVM and the 25 features selected by the DE/rand/1 optimised SVM radial kernel chosen for the DBN modelling and inference in the second stage because of their high accuracy and F1-score values.

Figure 6.3 shows the bar plot of the computed performance measures which indicates how the various optimisation algorithms performed on the dataset using different SVM kernels. From the figure it can be inferred that the PSO performed best overall on the linear kernel of the SVM while DE/rand/1 performed best on the radial kernel. The performance of the linear kernel over the radial was surprising in this study as the

radial kernel appears to used by most researchers however the this study also shows that datasets vary for different experiments and that various kernels of the SVM should be implemented when working on classification and feature selection tasks.

6.2.1 Results from Second Stage

With the features selected from the first stage, the second stage seeks to predict the relationship between the features by inferring the structure of a DBN. The DBN was inferred using the G1DBN algorithm [1] [131]. The modelling was based on the established assumption of this study that each patient (observation) represents a time point, which had been successfully implemented for diagnosis involving hypertension and cancer studies [3] [4]. The 64 cancer observations from the dataset and the selected features were used to infer which features at a previous time point X_{t-1}^{i} predicts a target feature at X_{t}^{j} at the current time point.

The G1DBN algorithm is implemented in two main steps. The first step involves the inference of a first-order dependence score matrix S_1 based on the Markov assumption that only the past variable which is one step back in time X_{t-1}^{i} predicts the variable at the current time point X_{t}^{j} by measuring the conditional dependence between the variables and any other variable X_{t-1}^{k}. Robust statistical estimators such as the M-estimators which include the Least Square (LS) estimator, Huber estimator and Tukey estimator are usually used [130]. In this study, the LS estimator was used to in the algorithm for inference of the relationships.

The algorithm works by computing for each $k \neq j$, the estimates $aij|k$ according to the LS estimator and the p-value $pij|k$ is derived from the standard significance test. A score matrix $S_1(i,j)$ is assigned to each potential edge $X_{t-1}^{i} \rightarrow X_{t}^{j}$ under the null assumption that $H_0^{ijk} : aij|k = 0$, which is equal to the maximum $Max_k \neq j(p_ij|k)$ computed p-values [1][130]. The most significant edge is represented by the smallest

score which contains the inferred directed acyclic graph DAG $G^{(1)}$ whose edge have a score below a chosen threshold $\alpha 1$. The G1DBN algorithm [1] is shown in Figure 6.4.

At the second step of the algorithm, a reduction in the search space of the inferred network is carried out. This is done using the score matrix S1 obtained from step 1 and an edge selection threshold $\alpha 2$ (from step 2) to infer the score S2 of each edge

Choose the Least Square estimator and set α1 and α2 thresholds

Step 1: Inferring $G^{(1)}$

$\forall i \in P$ (number of features)
$\forall j \in P$, $\forall k \neq j$, compute the p-value $p_{ij|k}$ under the null assumption.
$S_1(i,j) = \text{Max}_{k \neq j}(p_{ij|k})$
Set of edges $E(G^{(1)}) = \{ (X_i^{t-l}, X_j^t)_{t>1}; i,j \in P$, such that $S_1(i,j) < \alpha 1\}$

Step 2: Inferring G from $G^{(1)}$
If the maximum number of parents $N_{pa}^{Max}(G^{(1)}) \sim$ n - 1 degrees of freedom, chose a higher threshold α1 and go to step 1.
$\forall i$ such that number of parents $N_{pa}(X_i^{t-l}, G^{(1)}) \geq 1$, compute the p-value $p_{ij}^{(2)}$

$$S_2(i,j) = \begin{cases} p_{ij}^{(2)} \ \forall i,j \in P \ such \ that \ (X_i^{t-1}, X_j^t)_{t>1} \in G^{(1)}, \\ 1 \quad otherwise \end{cases}$$

Set of edges $E(G) = \{ (X_i^{t-l}, X_j^t)_{t>1}; i \in P \ (i,j) \in P$ such that $S_2(i,j) < \alpha 2\}$

FIGURE 6.4: The G1DBN Algorithm

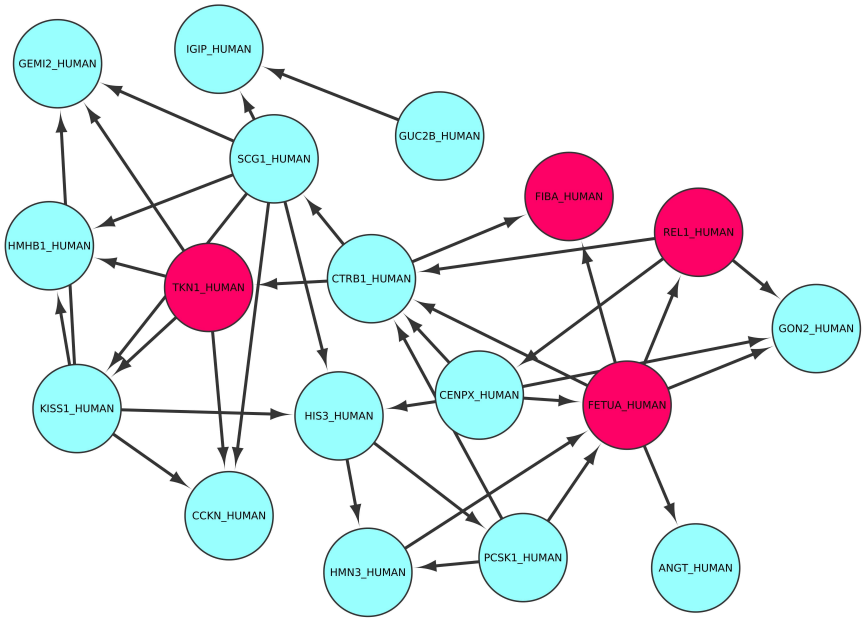

FIGURE 6.5: DBN model of the 18 features selected by the PSO-SVM-linear model showing significant interactions among high score edges. Red nodes are of key interest in this study.

of a DBN that describes full-order dependencies between successive variables. The algorithm is implemented in the G1DBN R programming language package [131] and is used to infer the temporal relationships of the top 18 and 25 features. The results of the inference with the algorithm may reveal novel pathways and relationships that have not been previously explored or studied. The inferred network shows arcs from parents to children with parents having high probability of inhibiting the corresponding gene depending on the strength of the arc. The network structure inferred shows predicted features at a previous time point $t-1$ pointing to a target feature at time t. More laboratory experiments which are needed for further clinical validation are beyond the scope of this thesis.

Figure 6.5 shows the DBN model of the 18 best features selected by the PSO linear. The network diagram made up of 36 edges was visualised using Cytoscape [18]. The inferred network shows that Alpha-2-HS-glycoprotein `FETUA_HUMAN` and tachykinin, precursor 1 `TKN1_HUMAN`, may be dominant hub genes. Also Alpha-2-HS-glycoprotein may be associated with Prorelaxin H1 `REL1-HUMAN` and Fibrinogen alpha chain `FIBA_HUMAN`. Research by [197] show that Alpha-2-HS-glycoprotein is a key biomarker which is believed to be associated with colorectal cancer and may also be aassocaited with early stages of breast cancer [198].

Studies by [199] also showed that Fibrinogen alpha chain may be a prognostic marker for colorectal cancer while tachykinin, precursor was found to be to have a sensitivity of 99% for non small cell lung cancer [200]. The result of this analysis also shows that Prorelaxin H1 `REL1_HUMAN` which in breast cancer research, controls that the in-vitro invasive potential [201] may be associated in time with Gonadotropin-releasing hormone II `GON2_HUMAN` which is known for being a key breast cancer biomarker [202] and in the prognosis of ovarian cancer [203], [204].

The inferred DBN of 24 out of the 25 features selected by DE/rand/1 optimised SVM-RFE radial kernel is shown in figure 6.5. These are the features that also had 97.3%

accuracy represented by 55 significant edges with one node pruned. Analysis revealed that the minor histocompatibility protein HMSD variant form `HMSDV_HUMAN` was a dominant hub in the network and had significant interactions with other features including with Pro-opiomelanocortin `COLI_HUMAN` which is a biomarker for small cell lung cancer [205]. Table 6.4 showed four features in common between the 18 and 25 selected features. Research shows that Proprotein convertase subtilisin/kexin type 1 is possibly associated with liver cancer [206] while Histatin-3 was found to be associated with oral candidiasis [207].

Molecular weight also plays a key role in colon cancer research as heavy kilo Dalton (kDa) protein biomarkers have been known to be associated with the disease. This includes metastasis associated in colon cancer-1 `MACC1_HUMAN` which has a molecular weight of 97kDa verified from UniProt [208] and was found to be highly associated with breast cancer and liver cancer (hepatocellular carcinoma) respectively [209], [210]. The E2F transcription factor 4 `E2F4_HUMAN`, another heavy protein biomarker

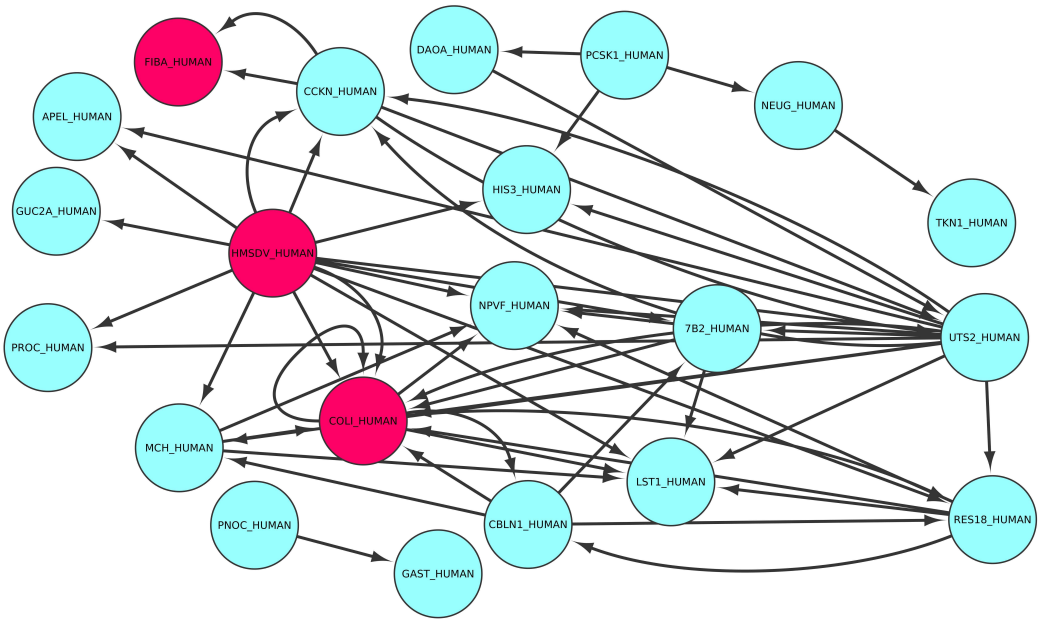

FIGURE 6.6: DBN model of 24 of the 25 features selected by the DE/rand/1 SVM-RFE with radial kernel showing key interactions among high score edges with one edge pruned. Red nodes are of key interest in this study.

with a molecular weight of 44 kDa, have also been linked to the metastasis of colorectal cancer [211].

6.3 Conclusion

This study aimed at selecting high quality colorectal cancer features using a more robust optimised two-stage approach. The hypothesis for the experiment states that optimised embedded methods of feature selection are better than non-optimised filter methods of feature selection in colorectal cancer diagnosis. The chapter also investigates biological interpretation of best colorectal protein features using two-stage bio-network discovery approach developed in chapter 5.

Six optimisation algorithms were adapted for parameter optimisation of the SVMRFE feature selection method which from literature is more accurate than other filter approaches used in previous studies. Three main kinds of SVM kernels, the linear, the polynomial and the radial basis function (RBF), were considered for feature selection using recursive feature elimination. The five DE algorithm variants and the PSO algorithm were used and implementation was done using the R Language for Statistical Computing.

The optimization algorithms were applied on the three different SVMRFE methods following the steps described in the presented pseudo-code. Cross-validation results show that the PSO's performance with the linear kernel of the SVM had the lowest cross validation error of 0.038 followed by the performance of the DE/rand/1 algorithm using the RBF kernel. The DE/rand-to-best/1 had the worst performance on average when compared with the other algorithms. Its performance on the polynomial and RBF kernels were 0.378 and 0.392 respectively with the performance on the linear kernel being 0.486. DE/best/1 had good performance on RBF kernel at 0.169 which was slightly better then the performance of DE/best/2 on the RBF kernel at 0.189. In

this study, even though the PSO achieved better results than the DE and its variants, the run time of DE was faster than that of the PSO. However, run time is only significant if the resulting solution is of good quality.

The optimised values of C and γ after cross-validation were used to selected best colorectal cancer features. The result showed that 18 and 25 features were respectively selected by the PSO and DE/rand/1 optimised SVMRFE methods. The features had the same highest average accuracy of 97.3% with F1-scores of 97.7 and 97.6 respectively. The temporal relationships of the selected features were modelled using the G1DBN algorithm where temporal interaction between a network predicted at a previous time point $t-1$ has a target feature at time t.

The 18 best features selected by the PSO linear were modelled using DBN. The resulting network was made up of 36 edges. The inferred network shows that Alpha-2-HS-glycoprotein `FETUA_HUMAN` and tachykinin, precursor 1 `TKN1_HUMAN`, may be dominant hub genes. Also Alpha-2-HS-glycoprotein may be associated with Prorelaxin H1 `REL1-HUMAN` and Fibrinogen alpha chain `FIBA_HUMAN`. Research by [197] show that Alpha-2-HS-glycoprotein is a key biomarker which is believed to be associated with colorectal cancer and may also be aassocaited with early stages of breast cancer [198].

The two-stage approach will aid in obtaining high quality feature with predicted interactions among features, however, one drawback is that the DBN algorithm adapted at the second stage of assumes a linear dependency among variables and hence modelled using a linear model which at the core of the algorithm with p-values as a measure of edge strength. This strong assumption of linearity is not always true in real biological systems which are inherently nonlinear. This leads to misrepresentation of actual dynamics of biological systems and chapter 7 aims to address this problem by proposing and successfully adapting nonlinear algorithmic methods.

TABLE 6.3: Overall performances of the optimisation algorithms for all SVM kernels showing the number of selected features and their performance measures

Optimization Algorithm	SVM-kernel	No. of features	Acc.	Sens.	Spec.	BCR	F1-Score	MCC
PSO	Linear	**18**	**0.973**	**1.00**	**0.938**	**0.969**	**0.977**	**0.946**
	Poly	37	0.878	0.818	0.867	0.842	0.857	0.675
	RBF	28	0.919	0.95	0.882	0.916	0.927	0.837
DE/rand/1	linear	12	0.811	0.727	0.933	0.83	0.821	0.651
	Poly	44	0.946	0.96	0.917	0.938	0.96	0.877
	RBF	**25**	**0.973**	**0.952**	**1.00**	**0.976**	**0.976**	**0.947**
DE/best/1	linear	51	0.865	0.947	0.778	0.863	0.878	0.738
	Poly	41	0.838	0.90	0.765	0.832	0.857	0.675
	RBF	55	0.919	1.00	0.786	0.893	0.939	0.834
DE/rand-to-best/1	linear	32	0.757	0.65	0.882	0.766	0.743	0.54
	Poly	24	0.838	0.714	1.00	0.857	0.833	0.721
	RBF	50	0.892	1.00	0.789	0.895	0.90	0.804
DE/best/2	Linear	85	0.892	0.909	0.867	0.888	0.909	0.7762
	Poly	53	0.946	0.909	1.00	0.955	0.952	0.896
	RBF	72	0.946	1.00	0.905	0.952	0.941	0.897
DE/rand/2	linear	39	0.946	0.895	1.00	0.947	0.944	0.897
	Poly	48	0.838	0.818	0.867	0.842	0.857	0.675
	RBF	**62**	**0.964**	**1.00**	**0.857**	**0.929**	**0.977**	**0.905**

TABLE 6.4: Four features selected in common between the 18 and 25 selected features

Swiss-Prot entry	Full name	Related disease	Reference
FIBA_HUMAN	Fibrinogen alpha chain	colon cancer	[199]
CCKN_HUMAN	Cholecystokinin	colon cancer	[212]
HIS3 HUMAN	Histatin-3	oral candidiasis	[207]
PCSK1_HUMAN	Proprotein convertase subtilisin/kexin type 1	liver cancer	[206]

Chapter 7

Inferring the Dynamics of Gene Regulatory Networks via Optimised Non-linear Predictors and DBN

7.1 Introduction

In chapter 6, ways to improve on the already developed two-stage DBN-based approach were proposed and implemented. In particular two major kinds of optimisation algorithms, particle swarm optimization and differential evolution algorithms, were adapted for parameter optimisation of the two-stage approach. The experimental results showed improved feature selection accuracy and more relevant features being selected. DBN-modelled relationships further revealed remarkable discovery of gene associations that might be important in the diagnosis of colorectal cancer and investigation of its metastasis. As noted however, the strong assumption of linearity of the DBN second stage does not fully capture the true representation of biological systems which are known to be nonlinear

The hypothesis for this chapter is that nonlinear DBN-based algorithms are better than linear DBN-bassed algorithms for the task of reverse engineering of gene expression network from time-course gene expression data. To test the hypothesis, two nonlinear DBN-based algorithms are developed and adapted to both simulated and real world data to prove their robustness. In particular, the methods developed are applied to inference of gene regulatory networks and ovarian cancer time course data. It is worth noting that the developed methods have successfully been published.

Inferring GRNs from high-dimensional time course data is a key challenge in bioinformatics. This is difficult because of the $p \gg n$ curse of dimensionality problem where the number of predictors p is much greater than the number of observations n. Various methods have been proposed in literature over the years for reverse engineering of GRN such as methods based on state-space models [213], [214], dynamic Bayesian network [1], [215], [125] and vector autoregressive models [216], [217], [218]. These methods however assume linearity of relationships among the genes which is not always a true representation of actual biological systems which are known to be inherently non-linear.

To address this problem, two dynamic Bayesian network-based algorithms based on non-linear predictors are proposed. This first algorithm is the Recurrent Neural Network Dynamic Bayesian Network (RNN-DBN) algorithm and the second is the Support Vector Regression Dynamic Bayesian Network (SVR-DBN) algorithm. These two developed algorithms were found to be better than exiting linear DBN-based G1DBN algorithm and were successfully published in IEEE conferences [5] [6]. The two DBN-based algorithms improve on the second stage (DBN) of the proposed two-stage optimisation approach introduced in chapter 5. The G1DBN algorithm was used to model and infer a DBN at the second stage but the algorithm assumes linear interaction among genes which made it less practical as biological networks are not always linearly related.

7.1.1 RNN-DBN

The RNN-DBN algorithm was proposed to overcome the limitations of linearity assumption in modelling GRNs which are inherently non-linear. The algorithm also overcomes the analytical intractability of methods based on Ordinary Differential Equation (ODE) such as the S-systems [219] and the non-linear dynamical system (NDS) [220], [221].

Figure 7.1 shows overall block diagram of the computational model adapted in this RNN-DBN subsection. It represents the RNN-DBN algorithm developed for inferring the dynamics of gene regulatory networks. It takes (time-course) gene regulatory network data as input. At the second step in the diagram, initial population of optimisation parameters are generated by the optimisation algorithm, in this study, both the inertia weight PSO and CLPSO were used. The parameters are the hidden layers and the learning parameters for the Elman RNN. The pseudo-code is also shown in Figure 7.2.

Within the DBN, the weights of the RNN are computed as probabilities p for each feature; this forms the score matrix M1. The maximum weight for each feature is then selected such that it is less than $\alpha 1$ threshold. The edges between the predicted M1 and the true matrix MT are set. The strength of these edges are determined by the weights. The mean squared error (MSE) between M1 and MT is computed and the optimisation algorithm varies the optimisation parameters until lowest MSE. The output is the vector of best parameters with lowest MSE.

With the optimised parameters, the weights of the nonlinear Elman RNN are computed as probabilities p within the DBN. The area under the precision-recall curve (AUPR) is computed using matrices M1 and MT. This is compared with the results from the standard G1DBN algorithm using Drosophila Melanogaster dataset and the result shown in Figure 7.3.

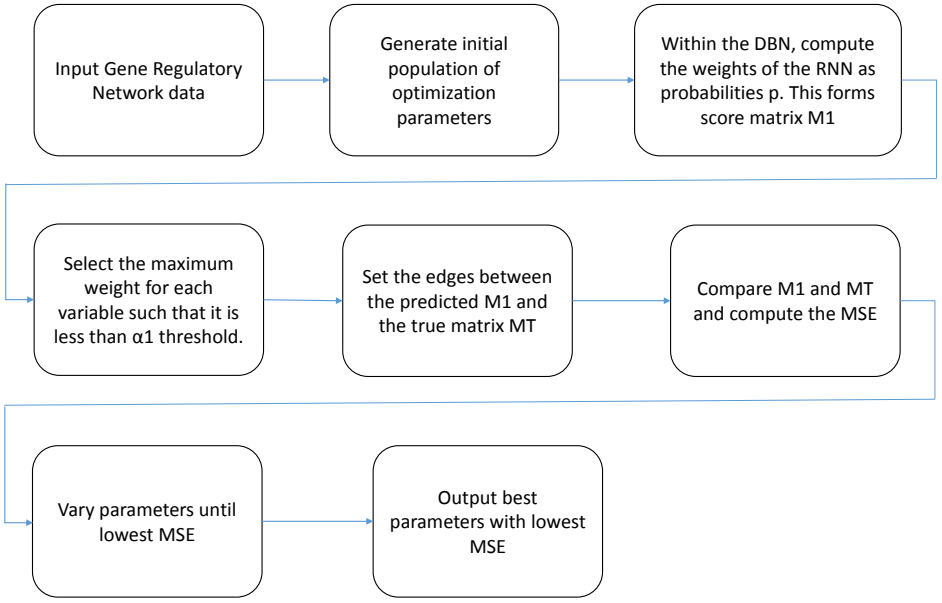

FIGURE 7.1: Block diagram showing the Computational Model of the Optimised RNN-DBN Algorithm [5]

Optimized RNN-DBN Algorithm

Generate an initial solution of N particles using PSO or CLPSO
while termination condition is not met **do**
//vary hidden layer size and learning parameter (the particles) using MSE as objective function.
for each particle i = 1,...,N **do**
$\forall i \in P$ (number of variables)
$\forall j \in P$, $\forall k \neq j$, compute the weights of the RNN as probability $p_{ij|k}$
Score matrix M1(i,j) = max($p_{ij|k}$)
Set of edges $E(G^{(1)})$ = { (X_i^{t-1}, X_j^t) $_{t>1}$; $i,j \in P$, such that $S_1(i,j) < \alpha 1$}
end for
end while

FIGURE 7.2: Pseudo-code of The Optimised RNN-DBN Algorithm

The DBN's learning approach is based on first-order conditional dependencies introduced by [1]. The contribution of the algorithm is improved modelling accuracy using optimized RNN. The Elman Recurrent Network was chosen because of it's simplicity and power [222], [223] in modelling non-linear feedback which are very common in biological systems. The state-space characteristics of the Elman Recurrent Network was another reason for its choice. It models non-linear relationships delayed by one

step back in time and using parameter optimization allows for solving specific inference problems [37]. The precision-recall (PR) curve was used as measurement metric and Figure 7.3 shows the RNN-DBN outperformed the G1DBN algorithm.

The inference of the algorithm is based on the Markov Assumption that only the past variable which is one step back in time predicts a target variable at the current time point by measuring the conditional dependence between the variables given any other variable. This is computed using the weights of the RNN. A score matrix M1(i,j) is assigned to each potential edge. The highest score indicates the most significant edge and the inferred DAG $G^{(1)}$ contains the edges which have been assigned a chosen

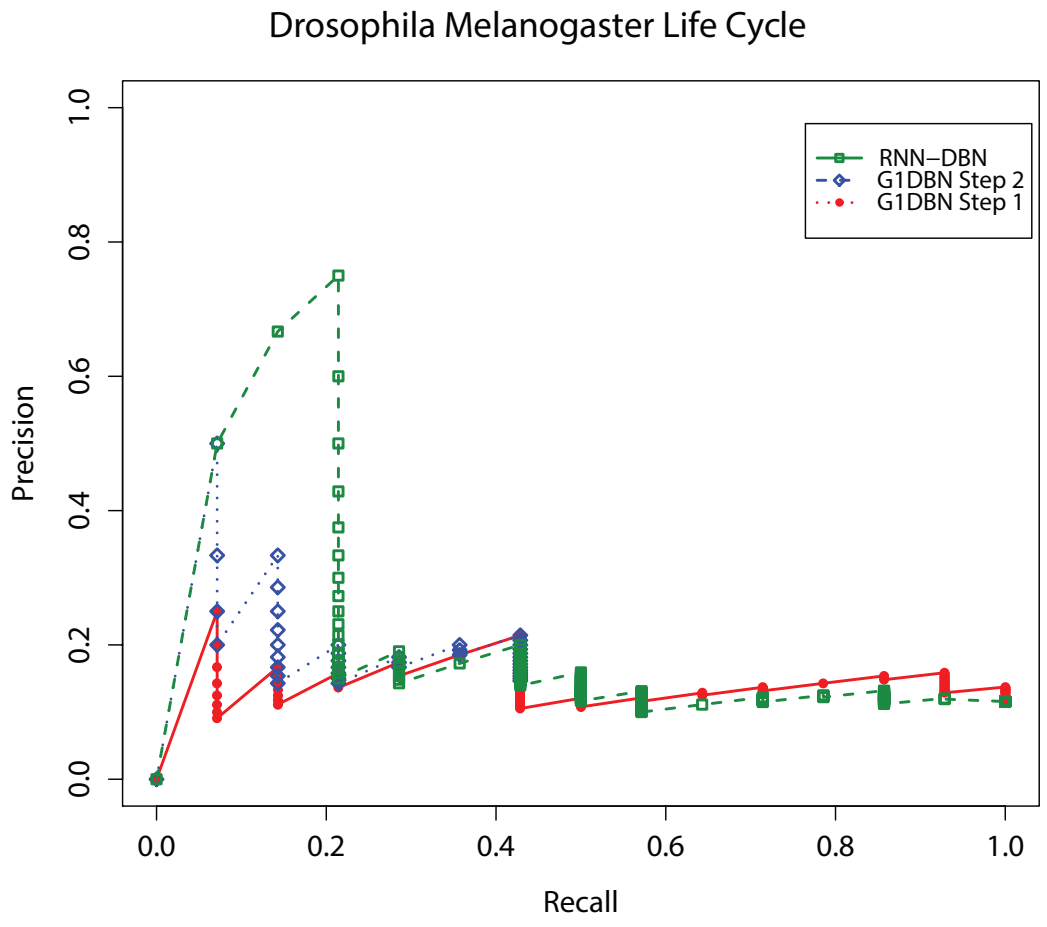

FIGURE 7.3: PR Curve Comparison of RNN-DBN and G1DBN on D. Melanogaster Benchmark dataset.

threshold $alpha1$.

7.1.2 Application Results of the RNN-DBN Algorithm

The algorithm was implemented using the R Statistical Language and the Elman Recurrent Network was computed using the RSNNS package [224]. The algorithm was demonstrated using real world time-course Drosophila Melanogaster benchmark dataset and human ovarian carcinoma time-course dataset. PSO and CLPSO were used to tune the parameters of the Elman Recurrent Network with the mean square error as the objective function. Inertia weight variant of the PSO was implemented. Parameters of the PSO and CLPSO used are: wmin=0.1, wmax=1, c1=0.4, c2=0.8, number of population=30, number of iteration=1000 and number of dimension (D)=4. The dimension is 4 because the parameters of the Elman Recurrent Network to be optimized are two hidden layers sizes and two learning parameters $\alpha 1$ and $\alpha 2$. Additional parameters of the CLPSO used are: PcMax=0.5 and PcMin=0.05.

7.1.2.1 Results based on Drosophila Melanogaster Dataset

The life cycle of the Drosophila Melanogaster dataset has been widely used in gene regulatory network studies and systems biology as its entire genome has been sequenced. In this study, the RNN-DBN algorithm was compared with the G1DBN developed by [1]. It has previously has been known to outperform other algorithms for inference such LASSO shrinkage, [1], the CLR [225] and the ARACNE [218] according to [213]. The RNN-DBN algorithm's performance was examined using 11 genes (variables) and 67 time points of Drosophila Melanogaster dataset. The dimension of the dataset was chosen as it has previously been used with the G1DBN R package [131]. Description of eight out of the 11 genes from the dataset used is shown in Appendix A.

The RNN was trained using the entire dataset to obtain optimum values for learning parameters $\alpha 1$ and $\alpha 2$ and also optimum values for the sizes of the hidden layers 1 and 2. The PSO and CLPSO were optimisation algorithms used to determine these optimum values and the better values were used. Table 7.2 shows the performance of the PSO and CLPSO algorithms. The CLPSO outperformed the PSO with lower MSE value of 0.0009 as compared to PSO's 0.0012 after 1000 iterations. The reason for using the entire dataset was that the resulting optimum parameters will be used by RNN within DBN algorithm for reverse engineering of gene regulatory network and not on a prediction test set.

This adaptation of RNN with a DBN is different from conventional training, validation and testing of artificial neural networks. In this adaptation, instead of using a linear model within DBN and computing the edge strength of the resulting network using p-values, the nonlinear RNN is used and the strength of the edges is computed using the weights of the RNN. The precision recall curve was used to compare the performance of both RNN-DBN and G1DBN algorithms.

Precision = TP / (TP + FP) Recall (Sensitivity) = TP / (TP + FN) where TP is the true positive, FP is the false positive, TN is the true negative and FN is the false negative.

Figure 7.3 shows that the RNN-DBN algorithm outperformed the G1DBN algorithm in modelling non-linear time-course GRN using real world data. It was more sensitive in modelling and representing non-linearities in biological systems such as GRN. The green lines represent the RNN-DBN, the blue lines represent the second step of the G1DBN and the red lines represent the first step of the G1DBN. The first step of the G1DBN involves inference of first-order conditional dependencies and the second step involves inference of full-order conditional dependencies.

At the first step and second step, the G1DBN attained maximum precisions of 0.2595 and 0.5214 respectively however the RNN-DBN outperformed it with a maximum precision of 0.7562. Table 7.1 shows the values of the area under the precision recall

curve (AUPR) which indicates the overall performance of the algorithms. It shows overall performance of 0.2166 for the RNN-DBN, 0.1404 for the second step of the G1DBN and 0.0881 for the first step of G1DBN algorithm. It can therefore be inferred from the results that the nonlinear RNN-DBN is more sensitive in inferring the structure of Drosophila Melanogaster than the linear G1DBN and hence can better model the network structure of time-course gene expression datasets in diagnosis of diseases such as cancer.

TABLE 7.1: Performance Comparison of RNN-DBN and G1DBN

Algorithm	**AUPR**
RNN-DBN	0.2166
G1DBN Step 2	0.1404
G1DBN Step 1	0.0881

7.1.2.2 Results based on Ovarian Carcinoma time-course dataset

The ovarian carcinoma time-course dataset GSE8057 [22] was downloaded analysed. It comprised of 51 time series arrays of Affymetrix HGU95Av2 GeneChips and 12625 genes. Further details about the data and the selection criteria for the genes used in this study can be found in section 4.5. The RNN-DBN algorithm was used to model the temporal relationships of the 34 genes used and time dependencies between the genes and connecting edges was visualised using Cytoscape [18]. Figure 7.7 shows 67 most significant edges above the $\alpha 1$ threshold with 30 genes realised from the inferred network.

Five potential hub genes each with four outgoing edges were discovered. These are These are flap structurespecific endonuclease 1, CDC6 cell division cycle 6 homolog (S. cerevisiae), kinesin family member 11, H2bd, TRAF family member-associated NFIB activator and histone 1. The hub genes discovered should be investigated further as the may be potential ovarian cancer biomarkers. For instance in DNA repair, the protein encoded by flap structure-specific endonuclease 1 removes 5 inches over

TABLE 7.2: Performance Comparison of RNN-DBN and G1DBN

Optimization Algorithm	**Hidden Layer 1**	**Hidden Layer 2**	$\alpha 1$	$\alpha 2$	**MSE**
PSO	11	8	3.1972	0.0443	0.0012
CLPSO	9	9	3.4162	0.0443	0.0009

hanging flaps; as a tumour suppressor, it plays essential role in tumorigenesis and studies by [226] found it to be a key gene in human breast carcinogenesis.

kinesin family member 11 (KIF11), also known as Eg5 might be a significant biomarker in pancreatic and lung cancer cells where dimethylenastron inhibits its function and further prevents the growth of the cancer cells. Eg5 is a mitotic kinesin that plays a crucial role in the formation of bipolar mitotic spindles by hydrolyzing ATP to push apart anti-parallel microtubules [227]. Further more, abnormal expression of both cyclin D1 and CDC6 by YB-1 may significantly contribute to lung carcinoma because CDC6 cell division cycle 6 homolog (S. cerevisiae) is essential for the initiation of DNA replication [228]. The significant interactions of these genes with other hub and non hub genes in the inferred network may reveal novel insights into the mechanism of the disease.

Stratified analysis of the discovered potential hub genes further showed that the expression levels of histone 1 was raised by oxaliplatin but remained constant with cisplatin while the expression level of TRAF family member-associated NFB activator was raised by cisplatin but remained constant with oxaliplatin. Also the expression levels of flap structure-specific endonuclease 1, kinesin family member 11 and CDC6 cell division cycle 6 homolog (S. cerevisiae) were decreased by oxaliplatin but remained constant with cisplatin platinum drugs.

Further analysis from Figure 7.7 also showed that three genes were highly regulated by other genes as shown in Table 7.3. This means that they had interactions coming to them rather than from them (as seen in the hub genes). The highly regulated genes are cyclin-dependent kinase inhibitor 1A (p21, Cip1) (CDKN1A), prostate differentiation factor (PLAB) and stratifin (SFN).

The five hub genes and three highly regulated genes may reveal novel insights into the disease metastasis and further clinical experiments are required as they may be highly

TABLE 7.3: Specific Hub Genes and Key Highly Regulated Genes

Probeset	**Gene Symbol**	**Gene Title**
Hub Genes		
39742_at	TANK	TRAF family member-associated NF-κB activator
38576_at	HIST1H2	histone 1, H2bd
1536_at	CDC6	CDC6 cell division cycle 6 homolog (S. cerevisiae)
40726_at	KIF11	kinesin family member 11
1515_at	FEN1	flap structure-specific endonuclease1
Regulated Genes		
2031_s_at	CDKN1A	cyclin-dependent kinase inhibitor 1A (p21, Cip1)
1890_at	PLAB	prostate differentiation factor
33322_i_at	SFN	stratifin

associated with ovarian cancer. The strongest edge in the inferred network was between cyclin-dependent kinase inhibitor 1A (p21, Cip1) and CDC6 cell division cycle 6 homolog (S. cerevisiae) (CDC6) followed by the edge between prostate differentiation factor (PLAB) and replication factor C (activator 1) 3, 38 kDa (RFC3).

7.2 Ensemble SVR-DBN

The ensemble SVR-DBN algorithm was proposed to overcome the limitations of linearity assumption in modelling GRNs. It involves the use of the non-linear radial basis function (RBF) kernel of the support vector regression algorithm within a dynamic Bayesian network framework. The DBN model is based on the first order conditional dependencies introduced by [1].

Figure 7.4 shows overall block diagram of the computational model adapted in this ensemble SVR-DBN section. The input is a time-course gene regulatory network data. The initial population of optimisation parameters are generated using the CLPSO

optimisation algorithm. Here the parameters to optimise are the C and γ of the SVR. Within the DBN, the weights of the SVR are computed as probabilities p. This forms the score matrix M1.

The maximum weight of the SVR for each feature is selected such that it is less than $\alpha 1$. This weight determines the strength of the edges of the DBN. The score matrix M2 based on $\alpha 2$ is evaluated as the full-order conditional dependencies of the DBN. This M2 is compared with true matrix MT and the inverse of the area under the precision-recall (1/AUPR) curve computed. The inverse is computed because it is a minimisation problem. The CLPSO varies the parameters until lowest 1/AUPR. The output is the inverse of the lowest (best) 1/AUPR. The contribution of the SVR-DBN algorithm is the improved sensitivity of inferring non-linear GRNs using non-linear RBF kernel of the SVR. The C and γ parameters of the SVR which represent a 2-Dimensional particle are also optimised using the comprehensive learning particle swarm optimisation (CLPSO) which was chosen because it yielded better results than the inertia weight PSO algorithm.

The G1DBN algorithm is implemented in two main steps. The first step involves the

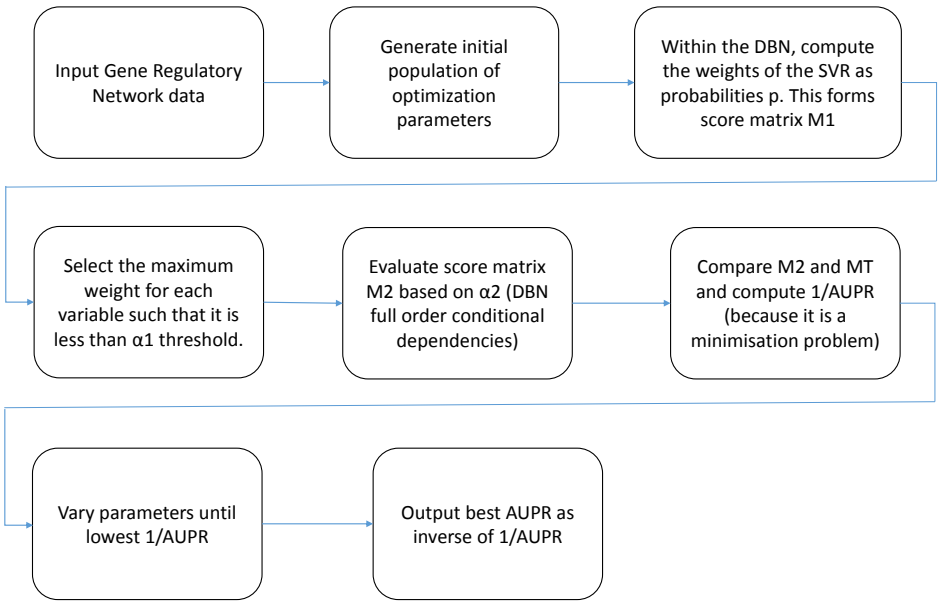

FIGURE 7.4: Block diagram showing the Computational Model of the Ensemble SVR-DBN Algorithm

inference of a first-order dependence score matrix S_1 based on the Markov assumption that only the past variable which is one step back in time X_{t-1}^{i} predicts the variable at the current time point X_t^j by measuring the conditional dependence between the variables and any other variable X_{t-1}^{k}. Robust statistical estimators such as the M-estimators which include the Least Square (LS) estimator, Huber estimator and Tukey estimator are usually used [130]. In this study, the LS estimator was used to in the algorithm for inference of the relationships.

The algorithm works by computing for each $k \neq j$, the estimates $aij|k$ according to the LS estimator and the p-value $pij|k$ is derived from the standard significance test. A score matrix $S_1(i,j)$ is assigned to each potential edge $X_{t-1}^{i} \rightarrow X_t^j$ under the null assumption that $H_0^{ijk} : aij|k = 0$, which is equal to the maximum $Max_k \neq j(p_i j|k)$ computed p-values [1][130]. The most significant edge is represented by the smallest score which contains the inferred directed acyclic graph DAG $G^{(1)}$ whose edge have a score below a chosen threshold $\alpha 1$.

At the second step of the algorithm, a reduction in the search space of the inferred network is carried out. This is done using the score matrix S1 obtained from step 1 and an edge selection threshold $\alpha 2$ (from step 2) to infer the score S2 of each edge of a DBN that describes full-order dependencies between successive variables.

The performance of the algorithm was proven on ten well-known benchmark datasets comprising of eight insilico-generated and two real world datasets. The insilico-generated datasets are made up of three Yeast knock-out gene expression square matrix datasets of sizes 10, 50 and 100 nodes. These were from the popular Dialogue for Reverse Engineering Assessments and Methods 3 (DREAM 3) datasets [67], and five DREAM 4 datasets of ten nodes and 21 time points each [68].

The two real world datasets used are the 11 genes and 67 time points of the Drosophila Melanogaster life cycle and the nine node network of the SOS DNA repair network of

Escherichia coli (E.coli).Miroslav Radman introduced the term "SOS response" to describe the network of the first experiment that supported the existence of an inducible DNA repair network [172]. The CLPSO was used to optimise the C and γ parameters of the SVR using the inverse of the area under the precision-recall (AUPR) as the objective function to be minimised. The parameters of the CLPSO were set at wmin=0.1, wmax=1, c1=0.4, c2=0.8, PcMax=0.5, PcMin=0.05, number of population=30, dimension (D) of population=2, number of iteration=1000.

As shown from the algorithm in Figure 7.5, the optimised ensemble SVR-DBN algorithm starts by generating an initial solution of N particles. The CLPSO was used in this study as it showed better performance however the PSO can also be used and may perform differently on different datasets. While the desired number of iterations is not yet met, particles of the CLPSO are constantly varied as the result of the DBN inference matrix M2 is constantly compared with that of the true matrix MT using the inverse of the Area under the precision recall curve as the objective function. The particle here are the C and γ parameters of the support vector regression algorithm which performs nonlinear inference within the DBN and the strength of the inferred edges is the weight of the SVR. The higher the weight, the higher strength of the edges between a predicted gene at time $t-1$ and a target gene at time t.

7.2.0.1 Results based on DREAM3 AND DREAM4 datasets

Table 7.4 shows the results obtained after 1000 iterations of the SVR-DBN algorithm for each of the datasets. The area under the precision-recall curve (AUPR) and the area under the receiver operating characteristics curves were computed for the SVR-DBN and the G1DBN. The G1DBN algorithm has been known to outperform other dynamic inference algorithms such as the LASSO, ARACNE, CLR and the VBSSM according to [213] and [70]. The result clearly shows the new SVR-DBN outperformed the

Optimized ensemble SVR-DBN

Generate an initial solution of N particles using PSO or CLPSO
while termination condition is not met **do**
//vary particles C and γ of the SVR using 1/AUPR of
//the DBN computation as objective function.
for each particle *i* = 1,...,N **do**
//this is the objective function
$\forall i \in P$ (number of variables)
$\forall j \in P$, $\forall k \neq j$, compute the weights of the SVR as probability $p_{ij|k}$
Score matrix M1*(i,j)* = max($p_{ij|k}$)
Graph $G^{(1)}$= (X_i^{t-1}, X_j^t) t>1; $i,j \in P$
$\forall i$ such that number of parents $N_{pa}(X_i^{t-1}, G^{(1)}) \geq 1$
compute the weights of the SVR as probability p_{ij}
//M2 improves on M1
Evaluate score matrix M2 as

$$M2(i,j) = \begin{cases} p_{ij} \ \forall i,j \in P \ s.t \ (X_i^{t-1}, X_j^t) t > 1 \in G^{(1)}, \\ 1 \quad otherwise \end{cases}$$

compare the matrix M2 with true Matrix MT
compute the AUPR using M2 and MT
return (1/AUPR)
end for
end while

FIGURE 7.5: Pseudo-code of The Ensemble SVR-DBN Algorithm [6]

G1DBN consistently with 12% average increase in total accuracy based on AUROC values over the 10 different datasets used in this study.

7.2.0.2 Results based on Real World Datasets

DREAM3 and DREAM4 are synthetically generated datasets and try to represent expression characteristics of real world datasets however they are generated using

TABLE 7.4: Comparison Results of SVR-DBN with G1DBN [6]

	G1DBN		Optimised SVR-DBN	
Dataset	AUPR	AUROC	AUPR	AUROC
DREAM4-1	0.1648	0.5537	**0.2287**	**0.4614**
DREAM4-2	0.1760	0.5476	**0.2303**	**0.6607**
DREAM4-3	0.1402	0.5153	**0.1663**	**0.5867**
DREAM4-4	0.1431	0.5561	**0.2607**	**0.7126**
DREAM4-5	0.1333	0.5483	**0.2175**	**0.6922**
DREAM3-10	0.1955	0.4862	**0.3323**	**0.6925**
DREAM3-50	0.0555	0.4831	**0.0856**	**0.5684**
DREAM3-100	0.0355	0.5353	**0.0402**	**0.5525**
D.Melanogaster	0.1113	0.5287	**0.1476**	**0.6282**
E.Coli	0.5649	0.4564	**0.7393**	**0.6931**

mathematical models such as ordinary differential equations or stochastic differential equations and are rare in practical biological scenarios. For example a single time series with 21 time points (sequence of 0-1000 by 50) used in DREAM4 are not very common in practice. For this reason, the SVRDBN algorithm was also tested using real world datasets. Two popular real world data commonly used are the *Drosophila Melanogaster* and the *Escherichia Coli*. Table 7.4 also shows that the SVR-DBN outperformed the G1DBN and is more sensitive and better in accuracy at capturing non-linearities of biological systems.

Figure 7.6 shows the ROC curves which indicates the performance of the algorithms on the two real world datasets used. The green line with open squares represents performance of the SVR-DBN on *Drosophila Melanogaster* while the red line with open squares represents performance of the G1DBN on *Drosophila Melanogaster*. It shows that the total area covered by the SVR-DBN at is more than that of the G1DBN. From Table 7.4, the area under the ROC curve is 0.5287 for the G1DBN and 0.6282 for the SVR-DBN. The area under the precision-recall curve (AUPR) is 0.1113 and 0.1476 for the G1DBN and the SVR-DBN.

The performance of the algorithms on *Escherichia Coli* dataset is also shown in Figure 7.6. The green line with closed circles represents the performance of the SVR-DBN and the red line with closed circles represent the performance of the G1DBN

algorithm. Again the area under the SVR-DBN ROC curve is more than that of the G1DBN and from Table 7.4, the G1DBN had 0.4546 while the SVR-DBN had 0.6931. Also the AUPR values of both the G1DBN and the SVR-DBN algorithms are 0.5649 and 0.7393 respectively. These results indicate clear out-performance of the nonlinear SVR-DBN algorithm over the existing G1DBN. It means that the SVR-DBN is more accurate in inferring regulatory network of time-course gene expression data and can better infer the structure of time-course gene expression data of diseases such as ovarian cancer. This can aid clinician in the development of new drug as the metastasis of the disease can better understood during prognosis.

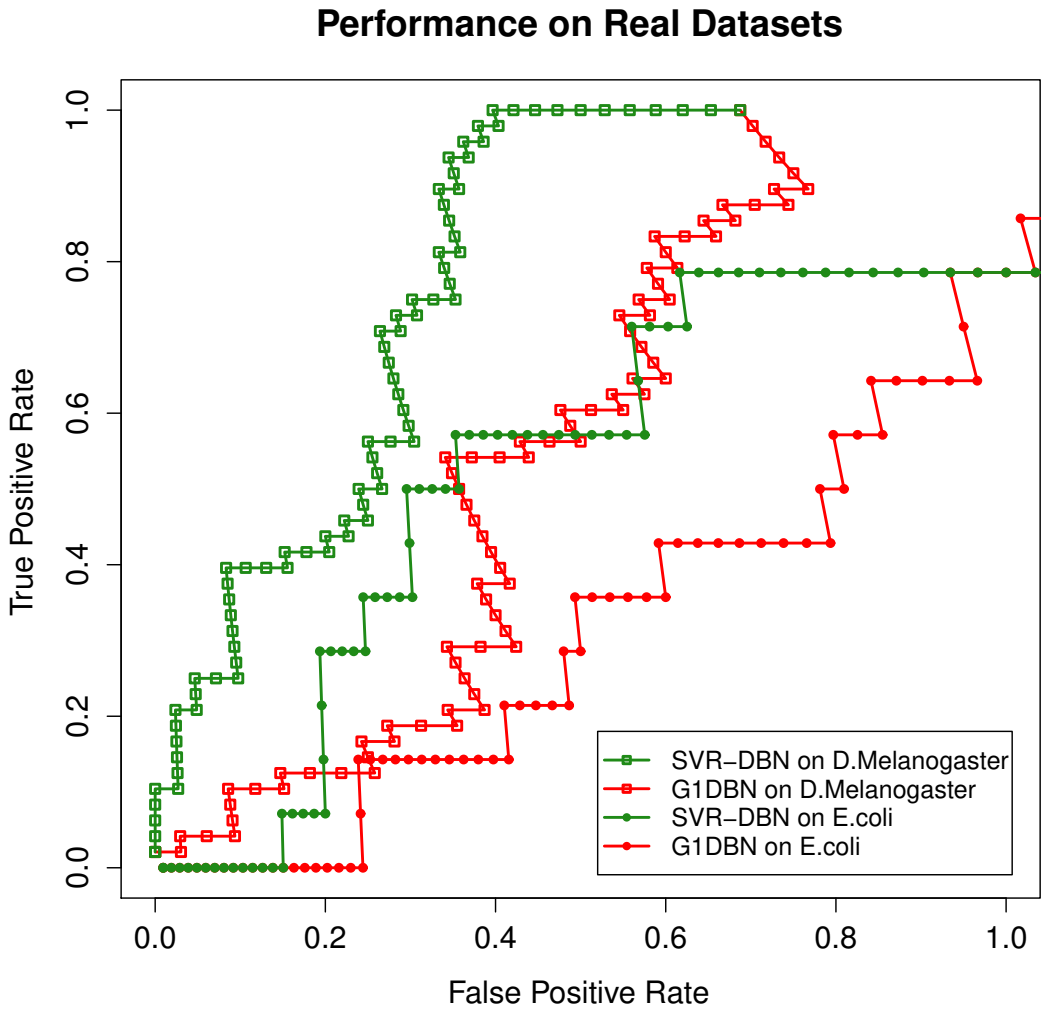

FIGURE 7.6: ROC curves showing performances of algorithms on the real datasets.

7.3 Conclusion

Modelling and inferring temporal associations of gene expression profiles can aid better understanding of molecular interactions which is important in drug discovery research. The dynamic Bayesian network is a robust and powerful inference algorithm preferred over static BN, the hidden Markov model and Kalman filters. The DBN modelling was extended to include inference based on vector autoregression and outperforms previous dynamic modelling algorithms. It was introduced and adapted as part of the two-stage approach proposed in this study and successfully developed using different case studies. One downside however was its rather unreal representation of biological networks as being linearly related to each other. The RNN-DBN and the SVR-DBN algorithms were proposed to address this problem and successfully developed and since published.

The two algorithms use nonlinear functions for nonlinear inference and allows for parameter optimisation for improved accuracy. In this study, the RNN-DBN was first developed before SVR-DBN. The stated hypothesis is that nonlinear DBN-based algorithms are better than linear DBN-based algorithms for the task of reverse engineering of gene expression network from time-course gene expression data. To test the hypothesis, two nonlinear DBN-based algorithms are developed and their efficiency tested using simulated and real world datasets.

The SVR-DBN was developed as an improvement to the RNN-DBN which was more complex and took more time to run. On the Drosophila Melanogaster dataset however, the RNN-DBN achieved AUPR value of 0.2166 while the SVR-DBN achieved a lower value of 0.1476. This was evident in this research; more runs of the RNN-DBN needed a more powerful computation platform. The SVM needs to consider the number of support vectors and the dimension of the dimension of the input space. The neural network however needs to consider the dimensions, the number of iterations, the number of hidden layers and the number of neurons in each hidden layer.

The results in this chapter appear to suggest that for the task of reverse engineering in gene regulatory networks, the more complex and more difficult RNN-DBN yields better results than the SVR-DBN. It will take multiple runs of both algorithms on all the ten datasets (and may be more) to conclude the better one. This is left as a further study. In terms of speed however, the SVR-DBN was much faster and could run on all the datasets easily.

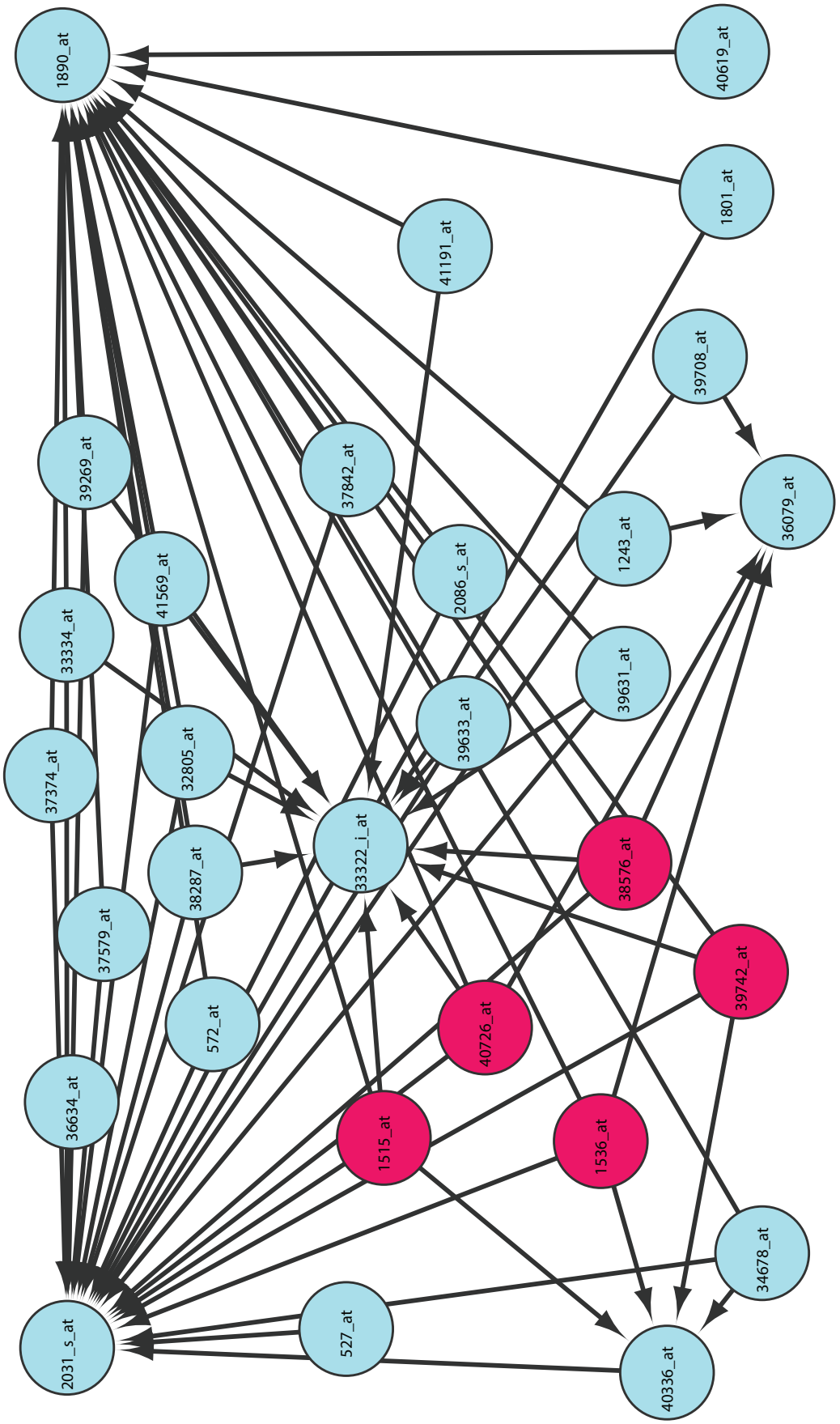

FIGURE 7.7: RNN-DBN inference model of time-course ovarian carcinoma data showing key hub genes (in red).

Chapter 8

Discussion and Conclusions

This thesis has been focused on covering research on using supervised learning methods and graphical models using dynamic Bayesian network for accurate feature selection and inference of relationships of biomedical data. Feature selection methods used include random forest recursive feature elimination, least absolute shrinkage and subset operator (LASSO) and support vector machine recursive feature elimination (SVMRFE) using the linear, the polynomial and radial basis function kernels. Various methods for inferring gene regulatory networks were reviewed.

This research introduced three main new methods. First, a two-stage bio-network discovery approach to identify and infer relationships between potential high quality biomarkers that may be of importance in clinical trial and drug discovery. At the first stage of the two-stage process, feature selection is carried using various feature selection methods mentioned and at the second stage, dynamic Bayesian network is used to infer the relationships among these features. The two-stage method was further improved by parameter optimisation using Particle Swarm Optimisation and Differential Evolution algorithms. The second method introduced in this thesis is the use of the RNN-DBN algorithm for reverse engineering of gene regulatory networks.

The methods developed are important and relevant both computationally and biologically. Computationally, a new two-stage framework was successfully developed and would help in better understanding temporal relationships between high quality selected features. The adapted optimisation algorithms means increased speed in delivering the tasks as best parameter values are automatically selected instead of manually trying out various values.

Biologically, the works done in the labs could be done faster using the proposed methods. They were tested on both real and simulated datasets and found to be more efficient than previous algorithms. Novel pathways that have not been discovered before in disease diagnosis can potentially be discovered by applying the proposed methods. The nonlinear inference DBN-based algorithms developed are more representative of actual dynamics of biological systems and would better capture the true picture for instance of temporal relationships among potential biomarkers, and help in the understanding of tumour metastasis and drug discovery efforts.

8.1 Summary of the Research Study

This study focused on developing Dynamic Bayesian Network (DBN) for analysis of high-dimensional biomedical data. This involved adapting various machine learning and computational optimisation methods. Machine learning methods adapted include random forest, the least absolute shrinkage and selection operator (LASSO) and three kinds of support vector machine: the linear, the polynomial and the radial basis function. The computational optimisation algorithms adapted are Particle Swarm Optimisation and its variant the Comprehensive Learning Particle Swarm Optimisation, and five variants of Differential Evolution algorithm which are: DE/rand/1, DE/best/1, DE/rand-to-best/1, DE/best/2 and DE/rand/2. The Differential Evolution and the Particle Swarm Optimisation algorithms were chosen because at the time of the PhD study,

they were not much explored in the DBN domain for biomarker discovery and were quite good at high dimension for parameter optimisation.

Dynamic Bayesian Network was chosen as the inference algorithm for time-course gene expression data because other alternative methods such as the Kalman Filter Model and the Hidden Markov Model (HMM) are not as suitable and efficient. For example, the Kalman Filter Model are not suitable for representing data with large qualitative discrete variables, and the Hidden Markov Model was more computationally complex than the DBN. As explained in chapter 1, it will take the HMM $O(TK^{2D})$ time for exact inference, where as it will take the DBN $O(TDK^{D+1})$ for exact inference for T sequence length with D chains and K number of states.

8.2 Overall achievements and contribution to literature

The main results and contribution of this thesis are summarised as follows:

- Development of a two-stage DBN-based bio-network discovery approach for analysing and inferring temporal associations of biomedical data.

  Accurate modelling of temporal relationships between biomarkers is important in disease prognosis and drug discovery. Existing methods focused only on classification performance of selected biomarkers but did not consider possible temporal relationships among feature subsets. At the first stage of the approach, feature selection is carried out using five different selection algorithms which are Random Forest Recursive Feature Elimination (RF-RFE), Least Absolute Shrinkage and Selection Operator (LASSO) and Linear, Polynomial and Radial Basis Function kernels of the Support Vector Machine Recursive Feature Elimination (SVMRFE) algorithm. Performance of the selected biomarkers were evaluated using machine learning criteria such as Sensitivity, Specificity, Accuracy, False Positive Rate and Matthews Correlation Coefficient. At the second

stage, the temporal relationships of features with the best overall performance from the first stage are inferred using Dynamic Bayesian Network. The relevance of the feature interactions and inferred network model is verified from literature for each of the use cases studied. Two case studies that were developed and published to address this objective are: ovarian cancer [3] and hypertension [4]. This objective is also addressed in chapter 5.

- Development of optimised two-stage approach using swarm optimisation for parameter optimisation

  This study aimed to address the problem of inaccuracies in selection of key biomarkers of potential relevance to drug discovery by using optimisation algorithms to fine-tune the parameters of feature selection algorithms for improved accuracy. Two key case studies were investigated:

  A) Diagnosis of Colorectal Cancer. In this study, significant pathways in colorectal cancer metastasis were discovered using the proposed two-stage DBN-based optimization approach. Previous modelling techniques from literature adopted less accurate filter methods of feature selection and methods of parameter optimization for improved accuracy were not proposed nor implemented. Furthermore, temporal relationships among potential biomarkers which might reveal significant pathways in the spread of the disease were not considered. This study aimed to address the aforementioned problems by adopting three kinds of more efficient SVMRFE feature selection algorithm.

  The algorithms were further optimised using particle swarm optimisation and five differential optimisation algorithms. The best performing features from the algorithm were modelled using DBN. The resulting inferred network, verified from published literature, showed that Alpha-2-HS-glycoprotein was highly associated with Fibrinogen alpha chain which had been shown to be a possible biomarker for colorectal cancer. This is addressed in chapter 6 and published in IET Systems Biology [15]

- Development and implementation of optimised Recurrent Neural Network Dynamic Bayesian Network (RNN-DBN) algorithm for accurate inference of Gene Regulatory Networks (GRNs).

  The study aimed to address the problem of linearity assumed by most temporal inference methods and inefficiencies of parametric methods such as the S-systems model. After analytical reviews of different recurrent neural network models, the Elman Recurrent Neural Network was chosen due to its simple feedback connections which makes it powerful for nonlinear mapping. The parameters of the RNN were further optimized using particle swarm optimisation (PSO) and comprehensive learning particle swarm optimization (CLPSO) algorithms.

  Results from the area under the precision recall (AUPR) curve using benchmark Drosophila Melanogaster dataset showed that the algorithm outperformed existing G1DBN algorithm which had been know to outperform three other algorithms: the LASSO, the ARACNE and the CLR algorithms. The developed algorithm was further used to model time-course human ovarian carcinoma cell dataset. Five key hub genes with four outgoing edges each were discovered. These are flap structure-specific endonuclease 1, kinesin family member 11, CDC6 cell division cycle 6 homolog (S. cerevisiae), histone 1, H2bd and TRAF family member-associated NFIB activator. This study is presented in chapter 7 and published in IEEE [5].

- Development and implementation of optimised Support Vector Regression Dynamic Bayesian Network (SVR-DBN)

  This study aims to further address the assumption of linearity by many gene regulatory network inference models proposed in literature. The study improves on DBN for modelling GRNs by using a more efficient non-linear support vector regression (SVR) algorithm. The SVR-DBN algorithm is further improved by the use of particle swarm optimisation to fine-tune the parameters of the SVR. The

robustness of the algorithm was tested on eight popular benchmark Dialogue for Reverse Engineering Assessments and Methods (DREAM) datasets and two real world datasets of Drosophila Melanogaster and Escherichia Coli. The results based on computed area under the precision recall curve (AUPR) and area under the receiver operating characteristics curve (AUROC) showed that the algorithm outperformed the existing G1DBN algorithm on all ten datasets. This study is presented in chapter 7 and published in IEEE [6].

8.3 Limitations and Future Work

New horizons for future work are suggested by the developments made in this. These are given below:

- The first order Markov assumption is used to simplify modelling efforts in time series where the present depends only on the past variable. A second order Markov process can be possible with many parameters to compute. The development of such models may better capture better representation of system dynamics.

- The methods developed can be applied to streaming analytics which is currently trending with the emergence of big data techniques. How can the methods developed be robust enough to handle data streams and infer relationships using real time that only from static data? This would be an interesting way to forward this research beyond what has been done.

- This entire research was conducted using individual computer nodes. At other times multiple computers in the laboratory were used to run experiments concurrently however with petabytes of data generated regularly, this work can be further developed to run on big data platforms such as Apache Spark and Hadoop.

Another criticism is that the algorithms developed were mainly single-threaded and multi-threaded and faster methods could be developed.

- In terms of DBN-based inference algorithms and further developement, others such as Modular networks [229] and James-Stein Shrinkage [230] can be further developed and improved upon could be compared with the ones developed. Other regression based DBN algorithms could also be developed and compared with the SVR-DBN developed in this work. Code could be optimised for better speed by adapting multi-threaded approaches using more recent developed libraries in R language. More bespoke libraries could be developed also be developed.

- Web application interface of the algorithms developed could mean they can be used more easily. The author was working on developing web front-end for ease of use and a small prototype have been developed however more can be done.

Appendix A

Appendices

A.1 7 of the 39 Ovarian Cancer Metabolites selected by the LASSO

	V19	V29	V51	V62	V77	V89	V100
1	12339.00	597.42	10131.00	1938.10	3325.70	104930.00	3492.50
2	26461.00	1142.80	4249.50	30.65	2559.10	68223.00	6143.20
3	35497.00	954.73	12952.00	248.63	6276.00	52304.00	31985.00
4	17220.00	1188.90	12054.00	2677.40	7981.10	76646.00	19436.00
5	22673.00	1763.30	56387.00	5170.00	1369.70	35534.00	0.00
6	13378.00	1423.30	17851.00	4401.50	990.45	142440.00	0.00
7	21667.00	1049.20	13021.00	7066.40	1866.20	25610.00	0.00
8	33408.00	1399.20	15430.00	3383.70	1699.80	86068.00	0.00
9	30616.00	1369.80	10236.00	17.84	2656.80	525.68	22.93
10	15621.00	1478.30	3690.90	2753.00	6638.40	3112.40	310.12
11	31453.00	2294.70	1756.00	0.00	3953.60	82864.00	0.00
12	18323.00	564.89	13201.00	3479.10	6458.80	14294.00	13197.00
13	22373.00	1890.90	15662.00	259.25	7525.10	31238.00	0.00

14	25198.00	1142.10	4475.60	0.00	3809.80	2711.80	0.00
15	29950.00	209.07	130.00	171.67	3479.40	1511.80	0.00
16	20961.00	1882.90	41224.00	623.93	8497.60	54820.00	0.00
17	32192.00	1161.40	141.76	298.26	9910.10	33788.00	1176.00
18	3334.70	1992.00	10026.00	277.65	7569.20	28327.00	45.37
19	24928.00	607.97	12941.00	1162.40	4063.50	18947.00	12439.00
20	21854.00	1730.50	3161.50	3484.50	4844.10	31903.00	0.00
21	31383.00	761.19	0.00	190.33	354.14	23748.00	384.66
22	26699.00	2112.20	4336.80	110.02	4690.60	12522.00	0.00
23	27353.00	1487.30	582.59	2611.60	2646.00	19525.00	11250.00
24	20053.00	331.89	10404.00	0.00	1072.30	5908.40	0.00
25	18481.00	965.67	15780.00	1436.30	4097.10	3120.00	0.00
26	344.78	398.58	46630.00	1457.50	4087.70	63779.00	0.00
27	24695.00	693.14	0.00	789.52	4216.10	60746.00	17330.00
28	21175.00	831.90	0.00	1335.20	1744.90	1664.50	3085.40
29	32273.00	1613.70	2739.10	0.00	5091.80	54049.00	10466.00
30	36174.00	1380.80	1051.10	2252.60	2408.40	52448.00	16376.00
31	19350.00	1071.70	402.89	6681.60	8343.90	49141.00	4776.40
32	27248.00	798.76	4422.60	0.00	6838.40	102290.00	137.32
33	26190.00	360.48	11194.00	4737.20	4805.80	9780.60	20030.00
34	27279.00	327.95	1251.20	351.29	6432.20	176690.00	26444.00
35	17547.00	728.91	68.89	7470.20	3144.70	330.64	24255.00
36	29456.00	1094.20	3909.70	1147.00	10987.00	77960.00	0.00
37	18056.00	370.59	29967.00	1939.80	5467.10	36427.00	14301.00
38	29085.00	1221.80	15119.00	2404.10	7208.80	40024.00	15082.00
39	16914.00	1062.10	9473.70	19693.00	3866.80	31193.00	3786.10
40	12298.00	699.89	18994.00	8539.00	3894.30	58057.00	6796.60
41	15496.00	2010.90	11265.00	55.77	4097.50	66400.00	1366.70

42	15153.00	1611.80	104290.00	1346.50	3035.10	29605.00	0.00
43	7760.70	912.15	116060.00	493.62	3884.80	5993.10	3913.40
44	35724.00	1333.10	55740.00	637.00	5834.40	71639.00	0.00
45	29448.00	2494.60	1120.60	6058.70	4179.30	1441.20	28.81
46	25976.00	507.17	0.00	6275.20	5019.10	63593.00	17725.00
47	30633.00	1074.60	29.26	66.71	4385.10	30650.00	0.00
48	15997.00	827.78	0.00	91.77	12069.00	13763.00	3578.60
49	31394.00	2042.90	0.00	0.00	3077.60	185.00	0.00
50	25879.00	1380.10	675.77	11705.00	4098.90	15688.00	1705.30
51	13456.00	1025.40	30075.00	0.00	4118.80	5400.50	2305.50
52	8430.80	942.92	27.64	2375.70	12002.00	2818.20	16.69
53	242.63	729.48	52593.00	1780.90	7151.50	26045.00	25.37
54	667.56	2471.50	10339.00	2844.70	4209.70	9502.70	0.00
55	41134.00	2499.10	8114.80	0.00	4802.90	25239.00	0.00
56	28190.00	1687.40	8366.80	8464.40	4418.10	7764.60	0.00
57	33193.00	2048.20	0.00	0.00	3119.00	92.99	0.00
58	62810.00	1407.40	8769.80	2372.60	2053.90	6068.20	3418.50
59	3361.50	1818.80	39793.00	1276.50	11490.00	2757.20	0.00
60	30532.00	2994.80	31243.00	46.78	6201.00	7323.10	0.00
61	43399.00	1099.90	76487.00	1430.70	5492.40	51411.00	0.00
62	11552.00	560.78	8397.70	0.00	1726.60	4172.20	0.00
63	45942.00	1149.50	543.03	532.40	6822.40	19558.00	0.00
64	32153.00	1832.40	1505.20	0.00	5337.60	49991.00	0.00
65	13368.00	2142.70	38670.00	696.13	10499.00	37959.00	0.00
66	25612.00	2938.50	26077.00	712.14	8895.50	7677.20	0.00
67	36443.00	398.84	42883.00	181.29	6354.70	146230.00	0.00
68	34762.00	895.55	12074.00	3573.40	5431.20	32242.00	0.00
69	47203.00	173.52	464.00	1472.70	5266.10	20677.00	2575.00

70	38923.00	1900.70	10237.00	725.90	8635.60	36822.00	0.00
71	27836.00	1612.50	499.16	15923.00	3519.10	10411.00	0.00
72	35487.00	369.60	1124.90	279.09	5559.50	1995.30	2222.40

A.2 DESCRIPTION OF THE 101 hypertension features selected by the LASSO

	Row	Col	ProbeName	Gene.Symbol	Unigene.ID.v163.	Locus.ID	p.value
315	7	15	probe_687	null	Hs.440731	352046	0.01
358	7	85	probe_757	SATB2	Hs.412327	23314	0.03
386	8	32	probe_816	CRABP2	Hs.183650	1382	0.00
1039	19	55	probe_2071	SCRG1	Hs.7122	11341	0.01
1183	21	76	probe_2316	PEPP-2	Hs.370012	84528	0.01
1503	26	61	probe_2861	CSS3	Hs.165050	337876	0.05
1711	29	91	probe_3227	SEC61A2	Hs.368481	55176	0.00
1902	32	67	probe_3539	null	Hs.283271	-1	0.01
2192	37	21	probe_4053	null	Hs.371800	-1	0.01
2226	37	70	probe_4102	null	Hs.171885	-1	0.04
2298	38	106	probe_4250	null	Hs.198671	-1	0.02
2299	38	111	probe_4255	null	Hs.72307	-1	0.01
2505	41	101	probe_4581	null	Hs.370546	-1	0.11
2782	46	70	probe_5110	null	Hs.35090	-1	0.00
3499	58	44	probe_6428	MNAB	Hs.112227	54542	0.04
3506	58	55	probe_6439	null	Hs.79856	-1	0.00
3734	62	2	probe_6834	MPP4	Hs.63085	58538	0.00

3884	64	12	probe_7068	null	Hs.290255	-1	0.02
4075	66	110	probe_7390	null	Hs.346735	346850	0.02
4154	68	8	probe_7512	null	Hs.207092	-1	0.01
4232	69	2	probe_7618	null	Hs.116153	-1	0.01
4998	80	28	probe_8876	null	Hs.282862	-1	0.03
5109	82	6	probe_9078	MGC20781	Hs.237536	115024	0.00
5127	82	38	probe_9110	null	Hs.242822	-1	0.01
5179	83	6	probe_9190	null	Hs.378505	-1	0.01
5458	87	41	probe_9673	null	Hs.444491	-1	0.11
5545	88	53	probe_9797	null	Hs.306968	-1	0.03
5667	90	39	probe_10007	null	Hs.130465	-1	0.00
5771	91	89	probe_10169	null	Hs.26026	-1	0.02
5938	94	7	probe_10423	ADAM29	Hs.126838	11086	0.04
5959	94	43	probe_10459	null	Hs.437554	-1	0.06
6208	97	103	probe_10855	LDOC1	Hs.45231	23641	0.00
6428	101	29	probe_11229	FLJ38705	Hs.149740	286128	0.00
7009	109	104	probe_12200	null	Hs.133900	-1	0.06
7706	120	55	probe_13383	PANX3	Hs.99235	116337	0.03
7732	120	108	probe_13436	null	Hs.385784	-1	0.00
7923	124	18	probe_13794	CAST	Hs.434457	26059	0.08
8136	127	96	probe_14208	null	Hs.143715	-1	0.00
8451	132	66	probe_14738	KCNK7	Hs.175218	10089	0.00
8583	134	44	probe_14940	null	Hs.201184	-1	0.06
9113	142	39	probe_15831	FOXG1B	Hs.525266	2290	0.19
9254	144	39	probe_16055	FUT6	Hs.32956	2528	0.05
9407	146	100	probe_16340	null	Hs.62697	-1	0.03
9450	147	50	probe_16402	CEACAM3	Hs.11	1084	0.10
9735	151	108	probe_16908	RIG	Hs.278503	10530	0.00

9856	153	74	probe_17098	LZTFL1	Hs.30824	54585	0.00
9984	155	95	probe_17343	KIAA1202	Hs.380697	57477	0.00
10568	164	51	probe_18307	VTS58635	Hs.437035	91608	0.02
10621	165	31	probe_18399	null	Hs.100636	-1	0.00
10883	169	10	probe_18826	DRP2	Hs.159291	1821	0.03
11249	174	84	probe_19460	MIST	Hs.381351	116449	0.00
11368	176	65	probe_19665	RNASE3	Hs.73839	6037	0.05
11631	180	109	probe_20157	SDOS	Hs.6949	84309	0.01
11710	182	31	probe_20303	RBMY1A3P	Hs.449529	286557	0.08
11965	186	102	probe_20822	null	Hs.255678	-1	0.01
12002	187	61	probe_20893	null	Hs.255496	-1	0.08
12267	191	38	probe_21318	FLJ23129	Hs.49421	79819	0.02
12313	191	101	probe_21381	FLJ33207	Hs.146083	339855	0.04
12376	193	17	probe_21521	RFX2	Hs.100007	5990	0.00
12602	196	105	probe_21945	ADAM21	Hs.178748	8747	0.05
12786	199	103	probe_22279	null	Hs.443145	341460	0.01
13024	203	60	probe_22684	null	Hs.97553	-1	0.02
13467	210	40	probe_23448	ATP1A4	Hs.367953	480	0.01
13790	215	41	probe_24009	null	Hs.372092	-1	0.01
13903	217	3	probe_24195	PRLR	Hs.212892	5618	0.00
14618	228	50	probe_25474	APOBEC3F	Hs.337667	200316	0.01
14786	231	6	probe_25766	null	Hs.363495	-1	0.05
14801	231	30	probe_25790	STAR	Hs.440760	6770	0.02
14977	233	90	probe_26074	FOXQ1	Hs.297452	94234	0.04
15131	236	60	probe_26380	KIAA0645	Hs.435022	9681	0.04
15380	240	75	probe_26843	BNIP1	Hs.145726	662	0.01
15764	246	43	probe_27483	null	Hs.188836	-1	0.01
15958	249	54	probe_27830	null	Hs.97300	145788	0.06

16114	251	112	probe_28112	GCGR	Hs.208	2642	0.01
16298	255	30	probe_28478	FLJ30634	Hs.350065	148697	0.01
16375	256	45	probe_28605	STEAP2	Hs.408200	261729	0.02
16500	258	98	probe_28882	null	Hs.381151	79944	0.00
16870	265	25	probe_29593	null	Hs.319565	-1	0.01
16926	266	19	probe_29699	null	Hs.280357	-1	0.00
17146	269	75	probe_30091	FLJ31393	Hs.350816	219445	0.00
17392	273	66	probe_30530	ARPP-21	Hs.412268	10777	0.00
18340	288	24	probe_32168	MYO7B	Hs.154578	4648	0.02
18495	290	25	probe_32393	KCNC2	Hs.345757	3747	0.09
18591	291	74	probe_32554	CNN1	Hs.21223	1264	0.00
18629	292	34	probe_32626	null	Hs.332123	-1	0.02
18712	293	50	probe_32754	ZNF37A	Hs.446382	7587	0.04
19134	300	5	probe_33493	null	Hs.195391	-1	0.03
19165	300	58	probe_33546	null	Hs.279596	-1	0.00
19366	303	49	probe_33873	SLC4A5	Hs.321127	57835	0.01
19730	308	82	probe_34466	MGC45404	Hs.371135	257397	0.04
20013	313	41	probe_34985	null	Hs.372738	-1	0.01
20024	313	64	probe_35008	MEP1A	Hs.179704	4224	0.06
20209	316	58	probe_35338	null	Hs.199882	-1	0.01
20475	320	78	probe_35806	null	Hs.208550	-1	0.10
20562	322	27	probe_35979	FLJ23356	Hs.277431	84197	0.05
20899	327	97	probe_36609	null	Hs.372172	-1	0.01
20915	328	25	probe_36649	CKLFSF1	Hs.513657	113540	0.01
20976	328	109	probe_36733	FLJ12687	Hs.417062	79979	0.01
21056	330	40	probe_36888	null	Hs.429904	338987	0.05
21581	339	49	probe_37905	ALDH1A2	Hs.435689	8854	0.04
21614	339	97	probe_37953	null	Hs.432395	-1	0.09

A.3 12 of the 18 Colorectal Cancer Spectral Profiles selected by PSO SVMRFE Linear

	1	2	3	4	5	6	7	8	9	10	11	12
V1	1.08	2.49	3.29	1.14	2.46	3.25	4.05	1.19	2.45	50.46	5.40	2.58
V2	1.48	1.25	2.99	1.34	1.27	3.72	6.67	0.94	1.21	21.44	7.64	1.15
V3	0.56	0.70	1.96	0.31	0.80	1.91	8.49	0.15	0.58	22.64	6.57	1.50
V4	2.07	1.68	4.81	1.97	1.66	2.70	6.87	1.90	1.66	28.88	9.15	1.26
V5	4.62	1.45	11.02	4.57	1.37	4.65	6.47	4.30	1.42	34.91	7.53	2.28
V6	1.99	1.59	5.26	1.94	1.48	5.89	7.15	1.91	1.53	49.83	6.99	5.09
V7	1.92	1.39	5.34	2.09	1.35	3.65	4.77	2.17	1.39	76.22	6.49	2.91
V8	2.61	2.49	3.49	2.69	2.41	5.86	7.24	2.75	2.43	25.04	9.08	2.80
V9	1.07	1.62	3.81	1.18	1.58	2.70	7.30	1.16	1.62	61.28	12.52	1.23
V10	1.19	1.68	5.91	1.27	1.75	5.25	5.85	1.35	1.60	33.82	6.41	1.42
V11	0.83	1.41	3.75	1.02	1.31	3.03	3.53	0.93	1.42	27.83	4.29	1.75
V12	2.16	0.73	6.39	2.32	0.67	2.33	2.36	2.34	0.77	44.55	6.06	1.91
V13	2.57	0.37	5.79	2.53	0.32	0.83	3.00	2.79	0.41	37.43	9.52	1.88
V14	2.03	0.77	8.78	1.87	0.75	2.84	4.59	1.65	0.79	11.98	10.19	1.64
V15	3.78	1.13	14.36	3.55	1.12	4.05	5.05	3.52	1.10	27.94	7.10	2.26
V16	0.73	0.46	2.35	0.72	0.50	2.23	3.99	0.72	0.53	57.47	5.55	0.31
V17	2.01	1.20	6.96	2.03	1.18	5.92	4.07	1.97	1.13	43.33	5.44	2.25
V18	2.98	1.46	9.71	2.98	1.37	5.89	9.53	3.04	1.50	34.34	8.03	2.40
V19	4.67	1.09	7.72	5.06	1.11	9.04	3.92	5.21	1.02	25.73	6.17	3.44
V20	2.05	0.81	10.89	2.02	0.85	9.49	5.64	1.93	0.72	66.05	7.29	4.07
V21	4.64	1.33	15.33	4.77	1.27	4.46	2.03	4.82	1.40	2.89	9.02	2.34
V22	2.86	1.27	6.51	3.09	1.25	6.15	4.87	3.08	1.33	13.12	2.77	3.32
V23	1.89	0.49	7.04	1.70	0.51	4.84	4.84	1.50	0.51	30.45	7.30	2.26
V24	3.70	0.92	9.33	3.99	0.94	5.82	7.62	4.45	0.90	11.65	11.81	3.20

V25	3.74	3.49	10.46	4.10	3.54	5.11	7.33	4.12	3.36	50.54	9.10	2.43
V26	3.51	1.06	15.11	3.65	1.06	3.14	3.33	3.71	1.07	80.25	7.60	4.04
V27	1.96	3.32	7.16	2.37	3.36	4.88	4.76	2.75	3.24	42.74	2.77	1.81
V28	0.61	2.87	6.68	0.88	2.85	2.19	7.64	1.13	2.80	27.97	4.06	5.62
V29	1.44	2.52	5.94	1.60	2.36	4.01	5.10	1.48	2.48	57.71	6.20	3.92
V30	1.47	3.94	10.83	1.72	3.94	5.82	3.53	1.77	3.80	87.27	6.64	3.81
V31	1.12	5.04	11.29	1.08	4.97	5.81	3.09	1.14	4.94	56.08	5.44	3.44
V32	1.67	2.71	4.51	1.68	2.65	4.33	3.78	1.84	2.70	52.61	10.43	2.39
V33	1.38	5.08	6.85	1.91	4.95	2.46	2.51	2.11	5.05	67.04	3.33	4.71
V34	2.80	5.01	24.74	2.94	4.79	2.55	4.64	3.03	4.93	15.41	9.36	1.93
V35	0.61	1.40	7.99	0.62	1.42	3.89	10.29	0.73	1.34	31.62	6.04	3.77
V36	0.31	3.08	5.20	0.45	3.12	1.60	5.45	0.78	3.05	37.53	7.78	1.51
V37	2.30	2.41	11.03	2.65	2.29	4.81	3.37	2.73	2.40	33.82	10.86	2.64
V38	2.69	3.07	14.72	3.13	2.96	7.93	4.07	3.49	3.15	19.35	13.28	2.17
V39	2.65	5.87	14.74	2.64	5.79	6.08	4.32	2.57	5.89	12.37	11.60	2.39
V40	2.11	4.34	13.97	2.34	4.07	4.14	3.25	2.60	4.42	19.98	7.38	2.32
V41	3.35	2.58	14.62	3.68	2.54	6.23	3.54	4.09	2.58	19.23	8.81	4.41
V42	1.50	6.43	5.77	1.59	6.42	8.51	5.94	1.70	6.42	15.75	5.36	4.18
V43	2.07	2.38	6.66	1.99	2.32	7.03	4.19	2.17	2.50	28.45	7.05	2.85
V44	1.56	2.42	7.60	1.75	2.28	5.12	4.83	1.73	2.49	39.53	3.06	2.29
V45	0.73	0.83	7.54	0.98	0.94	4.42	4.60	1.06	0.78	36.56	2.99	3.81
V46	2.26	1.00	7.91	2.29	0.92	2.25	3.59	2.38	1.04	71.69	8.73	3.06
V47	2.08	1.49	9.03	2.26	1.50	3.26	3.06	2.43	1.51	60.25	4.33	2.38
V48	2.59	2.03	12.81	2.59	2.01	2.21	4.29	2.55	2.01	64.54	1.44	3.37
V49	1.76	3.00	9.50	2.03	2.87	5.47	3.33	1.81	2.97	15.57	5.31	2.50
V50	1.28	4.32	3.69	1.33	4.26	5.40	4.27	1.21	4.27	28.58	5.42	1.66
V51	1.97	2.23	8.15	2.07	2.13	5.57	4.91	2.01	2.43	14.54	6.71	1.46
V52	2.36	1.28	8.53	2.36	1.24	6.28	4.80	2.34	1.26	9.03	6.46	2.30

V53	1.81	2.66	11.61	1.49	2.69	5.79	3.71	1.64	2.59	19.63	4.08	2.84
V54	2.20	5.76	11.80	2.22	5.72	7.27	4.10	2.14	5.70	9.16	5.55	3.58
V55	2.61	3.83	8.14	2.72	3.67	11.69	5.69	2.75	3.82	8.89	9.06	3.43
V56	2.72	2.94	7.80	2.98	3.03	9.82	6.31	3.13	2.84	19.78	6.72	2.80
V57	2.73	3.50	9.46	2.86	3.50	6.14	7.41	2.98	3.58	8.82	5.28	1.47
V58	4.53	2.62	16.68	4.68	2.56	3.82	3.54	4.94	2.62	22.11	3.81	1.63
V59	2.86	5.00	11.50	2.78	4.93	4.00	4.07	2.68	4.96	5.77	6.33	2.00
V60	0.57	1.61	4.69	0.49	1.52	9.36	10.34	0.53	1.53	33.00	4.77	1.20
V61	3.41	4.38	5.68	3.69	4.28	6.40	29.20	3.76	4.22	13.27	10.50	4.43
V62	5.18	3.16	12.26	5.35	3.11	6.69	14.71	5.22	3.02	10.90	9.17	2.16
V63	4.59	2.87	18.77	4.71	2.97	7.09	3.70	4.89	2.81	14.17	7.35	1.35
V64	4.65	1.21	10.82	4.89	1.22	7.91	6.74	4.77	1.21	28.09	5.79	4.29

A.4 Description of the Drosophila Melanogaster Dataset

	eve	gfl.lmd	twi	mlc1	sls	mhc	prm	actn
E01hunfertilized	-1.94	-0.31	-0.89	-2.47	-0.30	-2.96	-1.80	0.25
E01h	-0.56	0.62	-0.94	-2.38	-0.64	-4.27	-3.12	-0.08
E0515h	-0.06	0.59	-0.50	-3.78	-0.25	-4.39	-3.28	-0.06
E012h	-0.29	0.62	-0.82	-4.47	-0.73	-3.79	-2.89	-0.03
E01525h	0.93	0.25	0.17	-3.14	-0.39	-4.02	-2.70	0.02
E023h	1.46	0.51	1.00	-3.51	-0.36	-3.54	-2.83	-1.30
E02535h	1.89	0.73	1.74	-3.78	-0.53	-3.39	-2.55	-1.26
E034h	2.10	0.16	1.78	-3.90	-0.38	-3.34	-2.60	-1.23
E03545h	2.23	-0.15	1.66	-3.93	-0.25	-3.55	-2.29	-1.44
E045h	1.94	-0.50	2.01	-3.59	-0.66	-3.60	-2.46	-1.43

E04555h	1.26	0.23	1.82	-3.79	-0.39	-3.43	-2.45	-1.16
E056h	1.21	0.22	1.73	-3.84	-0.72	-3.51	-2.46	-1.40
E05565h	-0.13	0.78	0.53	-3.89	-0.74	-3.27	-2.06	-1.63
E067h	-0.38	0.72	0.21	-3.93	-0.60	-2.75	-2.00	-1.78
E078h	-0.26	0.62	-0.12	-3.82	-0.71	-3.01	-2.23	-1.52
E089h	-0.01	1.10	0.22	-1.53	0.18	-2.63	-2.10	-1.57
E0910h	0.16	0.99	-0.67	-3.22	-0.11	-3.20	-2.13	-0.65
E1011h	-0.02	0.69	-0.09	-2.74	-0.41	-1.59	-1.35	-0.80
E1112h	0.00	0.03	-0.73	-1.36	-0.09	-0.44	-0.98	-0.03
E1213h	-0.14	0.13	-0.31	1.20	-0.01	1.83	0.48	-0.09
E1314h	0.03	0.15	-0.57	0.64	-0.00	1.24	-0.38	-0.60
E1415h	-0.14	0.23	-0.43	1.02	0.22	1.83	0.09	-0.26
E1516h	0.03	-0.13	-0.59	1.64	0.27	2.33	0.46	-0.31
E1617h	0.06	-0.20	-0.28	1.86	0.06	2.67	1.63	-0.13
E1718h	-0.42	-0.48	-0.38	2.27	0.23	2.80	1.99	-0.35
E1819h	-0.30	-0.53	-0.41	2.82	0.07	3.23	1.79	-0.39
E1920h	-0.10	-0.86	-0.55	3.13	0.48	3.27	2.26	0.39
E2021h	-0.01	-0.34	-0.40	3.29	0.20	3.47	2.50	0.33
E2122h	-0.08	-0.31	-0.78	3.22	0.23	3.40	2.44	0.26
E2223h	0.02	-0.79	-0.59	2.89	0.15	3.22	2.37	0.19
E2324h	-0.03	-0.89	-0.36	2.84	0.35	3.72	2.38	-0.09
L24h	-0.26	-0.28	-1.17	2.90	-0.26	2.06	0.01	0.77
L33h	0.08	-0.22	-1.01	3.33	-0.06	2.24	0.38	1.24
L43h	-0.08	0.70	-1.43	3.76	-0.11	2.78	0.76	0.24
L49h	-0.24	-0.51	-0.72	2.53	-0.21	1.96	0.11	1.45
L57h	0.03	0.14	-1.48	3.21	-0.01	2.88	1.12	0.07
L67h	0.31	0.02	-0.82	3.26	-0.06	2.43	0.09	0.81
L72h	-0.40	0.27	-1.23	3.96	-0.15	4.07	1.37	-0.81

L84h	0.42	0.55	-1.53	4.04	0.11	3.13	1.12	-0.44
L96h	0.01	0.28	-0.85	3.37	0.55	3.60	2.09	-0.28
L105h	-1.01	0.82	-0.66	2.11	0.26	-0.40	-0.31	-0.17
M0h	-0.55	0.62	-1.05	-1.31	-0.18	-1.67	-0.51	-1.88
M02h	-1.10	1.46	-0.44	-4.04	0.10	-4.13	-1.68	-0.83
M04h	-0.86	1.13	-0.21	-3.30	0.26	-4.06	-1.31	-0.07
M06h	-1.12	0.80	-0.28	-2.90	0.25	-3.96	-0.75	-0.55
M08h	-0.63	1.77	-0.31	-2.12	-0.22	-3.23	-0.66	-0.12
M10h	-0.56	0.95	-0.34	-1.75	0.26	-3.44	-1.59	-0.10
M12h	0.14	0.81	0.29	-1.49	-0.43	-3.10	-1.94	-0.23
M16h	-1.17	0.67	-0.06	-4.51	-0.08	-4.24	-1.99	0.24
M20h	-1.24	0.72	-0.16	-3.85	-0.03	-4.45	-2.30	0.38
M24h	-1.20	0.67	-0.06	-2.65	-0.14	-2.74	-2.16	0.85
M30h	-1.01	0.51	-0.39	-4.70	-0.12	-4.02	-2.36	1.43
M36h	-1.24	0.91	-0.61	0.97	0.25	0.17	0.10	0.55
M42h	-1.05	0.22	-0.37	0.98	0.17	-0.57	-0.99	1.63
M48h	-0.95	0.21	-0.87	1.18	0.33	0.84	0.11	1.55
M60h	-0.71	0.86	-0.76	3.06	0.62	2.78	0.92	1.35
M72h	-0.85	0.88	-0.99	4.95	0.72	4.19	2.65	1.11
M80h	-0.94	0.57	-0.87	4.70	0.89	3.99	2.30	1.24
M96h	-2.02	-1.05	-1.11	4.45	1.46	4.83	3.05	1.65
Am024h	-1.04	0.05	-1.25	4.14	-0.05	3.42	1.19	0.60
Am03d	-0.39	-0.03	-1.03	1.44	0.11	1.17	0.09	0.82
Am05d	-0.54	-0.66	-1.33	0.06	0.20	0.68	-0.08	0.86
Am10d	-0.49	-0.49	-1.43	-0.49	0.22	0.07	-0.38	0.62
Am15d	-0.96	-0.31	-1.37	-0.61	0.31	-0.04	-0.47	0.79
Am20d	-0.81	-0.54	-0.95	-0.91	0.27	0.41	-0.49	0.51
Am25d	-1.02	-0.62	-1.08	-0.62	0.08	0.29	-0.25	1.18

Am30d	-1.24	-0.61	-1.27	-0.78	0.21	0.02	-0.32	0.33

A.5 Nine node network of the SOS DNA repair network of Escherichia coli

	V1	V2	V3	V4	V5	V6	V7	V8	V9
1	0.91	-0.13	-0.14	0.19	0.29	-0.06	-0.08	-0.02	-0.03
2	0.21	0.38	-0.12	0.06	0.17	-0.09	0.04	0.12	0.08
3	0.02	-0.11	10.52	0.06	0.08	0.01	0.06	0.09	-0.07
4	0.10	-0.05	-0.27	0.14	0.18	0.15	0.07	-0.00	0.28
5	0.12	-0.10	0.06	0.32	2.15	0.14	-0.07	0.14	0.11
6	0.08	-0.19	-0.12	0.25	0.35	2.02	-0.07	-0.17	-0.02
7	-0.12	-0.05	-0.10	-0.11	-0.01	0.10	3.07	0.36	0.22
8	0.18	-0.18	0.04	-0.07	-0.03	-0.15	0.01	26.63	0.09
9	0.07	-0.13	0.07	0.08	0.30	0.05	-0.06	0.27	0.67

A.6 10 features of the 100 DREAM3-100 simulated dataset used in chapter 7

	G1	G2	G3	G4	G5	G6	G7	G8	G9	G10
1	0.88	0.73	0.51	0.76	0.46	0.55	0.39	0.36	0.46	0.08
2	0.06	0.05	0.41	0.09	0.28	0.33	0.40	0.24	0.45	0.02
3	0.93	0.06	0.43	0.12	0.28	0.34	0.44	0.28	0.40	0.02
4	0.86	0.81	0.00	0.81	0.41	0.52	0.43	0.41	0.43	0.07

5	0.80	0.88	0.01	0.01	0.36	0.56	0.46	0.34	0.49	0.07
6	0.82	0.81	0.48	0.77	0.01	0.09	0.36	0.11	0.47	0.03
7	0.86	0.83	0.52	0.87	0.44	0.00	0.44	0.39	0.40	0.00
8	0.94	0.81	0.54	0.83	0.29	0.36	0.00	0.27	0.38	0.02
9	0.86	0.87	0.58	0.79	0.45	0.54	0.39	0.03	0.41	0.05
10	0.85	0.78	0.51	0.80	0.15	0.25	0.42	0.47	0.02	0.81
11	0.83	0.78	0.57	0.79	0.49	0.44	0.49	0.35	0.42	0.00
12	0.74	0.88	0.49	0.79	0.52	0.52	0.46	0.35	0.47	0.05
13	0.89	0.74	0.46	0.82	0.44	0.49	0.46	0.43	0.40	0.00
14	0.84	0.87	0.54	0.85	0.38	0.44	0.37	0.44	0.37	0.00
15	0.86	0.85	0.52	0.85	0.39	0.47	0.44	0.43	0.41	0.05
16	0.86	0.82	0.48	0.87	0.46	0.43	0.44	0.37	0.36	0.00
17	0.84	0.77	0.53	0.82	0.42	0.49	0.38	0.46	0.39	0.09
18	0.86	0.85	0.58	0.88	0.46	0.49	0.40	0.41	0.42	0.04
19	0.88	0.84	0.57	0.81	0.43	0.48	0.39	0.35	0.47	0.00
20	0.87	0.82	0.53	0.77	0.45	0.50	0.43	0.44	0.40	0.01
21	0.87	0.79	0.41	0.82	0.46	0.54	0.42	0.38	0.88	0.00
22	0.85	0.87	0.54	0.80	0.47	0.49	0.38	0.38	0.41	0.05
23	0.80	0.91	0.50	0.86	0.44	0.48	0.41	0.37	0.44	0.05
24	0.86	0.85	0.52	0.82	0.51	0.49	0.46	0.44	0.48	0.08
25	0.85	0.89	0.52	0.86	0.45	0.45	0.43	0.42	0.42	0.01
26	0.82	0.80	0.55	0.85	0.52	0.44	0.48	0.32	0.42	0.00
27	0.87	0.88	0.56	0.80	0.45	0.45	0.47	0.41	0.45	0.00
28	0.86	0.82	0.56	0.85	0.50	0.53	0.38	0.40	0.40	0.00
29	0.77	0.77	0.51	0.84	0.38	0.49	0.44	0.36	0.41	0.10
30	0.85	0.82	0.47	0.88	0.47	0.49	0.43	0.40	0.53	0.05
31	0.85	0.88	0.53	0.83	0.40	0.44	0.37	0.30	0.47	0.00
32	0.84	0.83	0.50	0.80	0.39	0.41	0.45	0.40	0.39	0.08

33	0.86	0.83	0.52	0.87	0.44	0.41	0.41	0.38	0.45	0.09
34	0.91	0.84	0.57	0.83	0.44	0.52	0.42	0.36	0.41	0.09
35	0.87	0.85	0.52	0.85	0.44	0.47	0.34	0.40	0.50	0.11
36	0.85	0.82	0.55	0.81	0.52	0.43	0.44	0.42	0.50	0.02
37	0.80	0.79	0.57	0.83	0.43	0.42	0.46	0.45	0.39	0.05
38	0.86	0.89	0.53	0.78	0.43	0.47	0.41	0.42	0.50	0.11
39	0.84	0.86	0.56	0.85	0.46	0.46	0.45	0.38	0.50	0.07
40	0.76	0.81	0.54	0.84	0.46	0.48	0.34	0.40	0.46	0.00
41	0.89	0.84	0.52	0.80	0.44	0.49	0.48	0.38	0.46	0.10
42	0.88	0.84	0.54	0.82	0.44	0.49	0.43	0.39	0.47	0.04
43	0.88	0.82	0.63	0.82	0.49	0.43	0.49	0.40	0.50	0.00
44	0.78	0.84	0.62	0.80	0.40	0.43	0.39	0.34	0.36	0.05
45	0.90	0.84	0.51	0.78	0.47	0.44	0.42	0.36	0.47	0.00
46	0.84	0.78	0.50	0.83	0.45	0.36	0.39	0.41	0.50	0.05
47	0.80	0.77	0.52	0.77	0.48	0.48	0.35	0.43	0.40	0.05
48	0.90	0.87	0.53	0.84	0.49	0.45	0.36	0.39	0.43	0.00
49	0.83	0.73	0.48	0.79	0.41	0.46	0.50	0.33	0.41	0.01
50	0.84	0.85	0.58	0.87	0.45	0.46	0.41	0.35	0.49	0.02
51	0.89	0.79	0.54	0.75	0.40	0.47	0.43	0.33	0.48	0.04
52	0.89	0.81	0.47	0.83	0.44	0.48	0.93	0.38	0.35	0.02
53	0.82	0.81	0.53	0.80	0.49	0.43	0.50	0.34	0.36	0.00
54	0.92	0.76	0.54	0.80	0.50	0.49	0.39	0.33	0.41	0.10
55	0.81	0.86	0.53	0.83	0.44	0.47	0.42	0.31	0.38	0.00
56	0.86	0.89	0.52	0.86	0.48	0.52	0.78	0.38	0.56	0.13
57	0.78	0.77	0.43	0.82	0.50	0.48	0.44	0.37	0.44	0.08
58	0.92	0.87	0.51	0.84	0.42	0.48	0.45	0.42	0.40	0.11
59	0.85	0.89	0.59	0.84	0.47	0.45	0.42	0.39	0.36	0.00
60	0.88	0.84	0.54	0.78	0.43	0.55	0.44	0.42	0.52	0.00

61	0.82	0.85	0.57	0.90	0.45	0.46	0.43	0.37	0.44	0.02
62	0.84	0.89	0.53	0.82	0.50	0.49	0.60	0.45	0.46	0.05
63	0.87	0.83	0.52	0.86	0.40	0.49	0.43	0.32	0.38	0.00
64	0.83	0.83	0.56	0.90	0.46	0.44	0.41	0.51	0.42	0.00
65	0.88	0.78	0.53	0.79	0.41	0.46	0.42	0.32	0.44	0.05
66	0.80	0.82	0.49	0.88	0.43	0.48	0.49	0.31	0.52	0.07
67	0.83	0.84	0.51	0.80	0.41	0.48	0.44	0.38	0.42	0.02
68	0.82	0.91	0.44	0.84	0.46	0.49	0.39	0.33	0.47	0.01
69	0.91	0.82	0.52	0.87	0.45	0.43	0.41	0.33	0.37	0.09
70	0.86	0.81	0.56	0.92	0.38	0.43	0.39	0.36	0.44	0.00
71	0.82	0.86	0.56	0.89	0.45	0.54	0.41	0.45	0.48	0.01
72	0.84	0.89	0.50	0.82	0.47	0.52	0.39	0.32	0.43	0.07
73	0.89	0.83	0.53	0.80	0.44	0.55	0.42	0.36	0.52	0.00
74	0.87	0.85	0.52	0.83	0.46	0.53	0.38	0.42	0.42	0.00
75	0.83	0.84	0.50	0.79	0.36	0.44	0.48	0.38	0.40	0.05
76	0.83	0.85	0.59	0.78	0.48	0.51	0.40	0.44	0.45	0.00
77	0.86	0.80	0.52	0.81	0.44	0.52	0.45	0.41	0.42	0.05
78	0.90	0.77	0.49	0.82	0.49	0.49	0.42	0.35	0.40	0.05
79	0.80	0.84	0.48	0.78	0.42	0.49	0.44	0.42	0.44	0.04
80	0.90	0.89	0.45	0.84	0.40	0.46	0.47	0.41	0.34	0.07
81	0.89	0.77	0.50	0.86	0.46	0.55	0.40	0.37	0.46	0.00
82	0.89	0.88	0.48	0.91	0.47	0.47	0.40	0.41	0.46	0.01
83	0.91	0.84	0.60	0.83	0.48	0.40	0.38	0.41	0.38	0.02
84	0.91	0.77	0.49	0.89	0.40	0.48	0.41	0.37	0.50	0.04
85	0.92	0.91	0.53	0.86	0.36	0.43	0.49	0.39	0.40	0.03
86	0.82	0.82	0.51	0.83	0.48	0.47	0.44	0.31	0.46	0.03
87	0.93	0.76	0.57	0.82	0.42	0.45	0.44	0.36	0.44	0.06
88	0.82	0.78	0.57	0.80	0.38	0.46	0.44	0.38	0.41	0.06

89	0.87	0.89	0.52	0.78	0.44	0.47	0.47	0.31	0.40	0.01
90	0.93	0.80	0.58	0.86	0.46	0.44	0.35	0.43	0.43	0.00
91	0.85	0.83	0.55	0.88	0.39	0.42	0.45	0.37	0.44	0.08
92	0.87	0.82	0.59	0.87	0.40	0.44	0.39	0.41	0.42	0.00
93	0.89	0.76	0.54	0.83	0.43	0.47	0.38	0.34	0.43	0.00
94	0.85	0.81	0.54	0.82	0.47	0.45	0.43	0.33	0.44	0.15
95	0.85	0.76	0.56	0.82	0.43	0.43	0.42	0.41	0.41	0.00
96	0.79	0.74	0.46	0.90	0.41	0.46	0.43	0.38	0.43	0.01
97	0.93	0.77	0.61	0.86	0.44	0.49	0.42	0.46	0.44	0.03
98	0.89	0.75	0.54	0.85	0.50	0.49	0.43	0.40	0.39	0.00
99	0.82	0.80	0.53	0.80	0.47	0.42	0.38	0.45	0.48	0.00
100	0.83	0.67	0.47	0.85	0.40	0.50	0.37	0.41	0.43	0.06
101	0.78	0.73	0.53	0.82	0.42	0.52	0.41	0.36	0.44	0.02

References

[1] S. Lèbre, "Inferring dynamic genetic networks with low order independencies," *Statistical applications in genetics and molecular biology*, vol. 8, no. 1, pp. 1–38, 2009.

[2] Z.-P. Liu, "Reverse engineering of genome-wide gene regulatory networks from gene expression data," *Current genomics*, vol. 16, no. 1, pp. 3–22, 2015.

[3] A. Akutekwe and H. Seker, "Two-stage computational bio-network discovery approach for metabolites: Ovarian cancer as a case study," in *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, June 2014, pp. 97–100.

[4] A. Akutekwe, "A hybrid dynamic bayesian network approach for modelling temporal associations of gene expressions for hypertension diagnosis," in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Aug 2014, pp. 804–807.

[5] A. Akutekwe and H. Seker, "Inferring the dynamics of gene regulatory networks via optimized recurrent neural network and dynamic bayesian network," in *2015 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, Aug 2015, pp. 1–8.

[6] A. Akutekwe, "Inference of nonlinear gene regulatory networks through optimized ensemble of support vector regression and dynamic bayesian networks,"

in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug 2015, pp. 8177–8180.

[7] E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar, M. Doyle, W. FitzHugh *et al.*, "Initial sequencing and analysis of the human genome," *Nature*, vol. 409, no. 6822, pp. 860–921, 2001.

[8] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt *et al.*, "The sequence of the human genome," *science*, vol. 291, no. 5507, pp. 1304–1351, 2001.

[9] Z. Ibrahim, A. Ngom, and A. Y. Tawfik, "Using qualitative probability in reverse-engineering gene regulatory networks," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no. 2, pp. 326–334, 2011.

[10] M. Li, X. Wu, J. Wang, and Y. Pan, "Towards the identification of protein complexes and functional modules by integrating ppi network and gene expression data," *BMC bioinformatics*, vol. 13, no. 1, p. 1, 2012.

[11] N. Xuan, M. Chetty, R. Coppel, and P. P. Wangikar, "Gene regulatory network modeling via global optimization of high-order dynamic bayesian network," *BMC bioinformatics*, vol. 13, no. 1, p. 131, 2012.

[12] S. M. Hill, Y. Lu, J. Molina, L. M. Heiser, P. T. Spellman, T. P. Speed, J. W. Gray, G. B. Mills, and S. Mukherjee, "Bayesian inference of signaling network topology in a cancer cell line," *Bioinformatics*, vol. 28, no. 21, pp. 2804–2810, 2012.

[13] S. Russell and P. Norvig, *Artificial intelligence: a modern approach*. Prentice hall Upper Saddle River, 2010, vol. 2, ch. 15: Probabilistic Reasoning over Time.

[14] D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[15] A. Akutekwe, H. Seker, and S. Yang, "In silico discovery of significant pathways in colorectal cancer metastasis using a two-stage optimisation approach," *IET Systems Biology*, vol. 9, no. 6, pp. 294–302, 2015.

[16] A. Akutekwe, H. Seker, and S. Iliya, "An optimized hybrid dynamic bayesian network approach using differential evolution algorithm for the diagnosis of hepatocellular carcinoma," in *2014 IEEE 6th International Conference on Adaptive Science Technology (ICAST)*, Oct 2014, pp. 1–6.

[17] R. C. Team, "R: A language and environment for statistical computing," 2013.

[18] M. E. Smoot, K. Ono, J. Ruscheinski, P.-L. Wang, and T. Ideker, "Cytoscape 2.8: new features for data integration and network visualization," *Bioinformatics*, vol. 27, no. 3, pp. 431–432, 2011.

[19] W. Guan, M. Zhou, C. Y. Hampton, B. B. Benigno, L. D. Walker, A. Gray, J. F. McDonald, and F. M. Fernández, "Ovarian cancer detection from metabolomic liquid chromatography/mass spectrometry data by support vector machines," *BMC bioinformatics*, vol. 10, no. 1, p. 1, 2009.

[20] K.-S. Lynn, L.-L. Li, Y.-J. Lin, C.-H. Wang, S.-H. Sheng, J.-H. Lin, W. Liao, W.-L. Hsu, and W.-H. Pan, "A neural network model for constructing endophenotypes of common complex diseases: an application to male young-onset hypertension microarray data," *Bioinformatics*, vol. 25, no. 8, pp. 981–988, 2009.

[21] M. E. de Noo, A. M. Deelder, M. R. J. van der Werff, and R. A. E. M. Tollenaar, "Detection of colorectal cancer using MALDI-TOF serum protein profiling," *European Journal of Gastroenterology & Hepatology*, vol. 19, no. 10, p. A78, OCT 2007.

[22] Y. F. BRUN, R. VARMA, S. M. HECTOR, L. PENDYALA, R. TUMMALA, and W. R. GRECO, "Simultaneous modeling of concentration-effect and time-course patterns in gene expression data from microarrays," *Cancer Genomics-Proteomics*, vol. 5, no. 1, pp. 43–53, 2008.

[23] A. Akutekwe and H. Seker, "Two-stage bioinformatics approach for the diagnosis of hepatocellular carcinoma and discovery of its bio-network," in *International Conference on Applied Informatics for Health and Life Sciences (AIHLS)*, October 2014, pp. 91–94.

[24] R. H. Shumway and D. S. Stoffer, *Time series analysis and its applications: with R examples*. Springer Science & Business Media, 2010.

[25] R. Nagarajan, M. Scutari, and S. Lèbre, "Bayesian networks in r," *Springer*, vol. 122, pp. 125–127, 2013.

[26] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[27] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[28] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine learning*, vol. 46, no. 1-3, pp. 389–422, 2002.

[29] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 2013.

[30] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[31] A. Cutler, D. R. Cutler, and J. R. Stevens, "Random forests," in *Ensemble Machine Learning*. Springer, 2012, pp. 157–175.

[32] H. T. Kam, "Random decision forest," in *Proc. of the 3rd Int'l Conf. on Document Analysis and Recognition, Montreal, Canada, August*, 1995, pp. 14–18.

[33] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE transactions on pattern analysis and machine intelligence*, vol. 20, no. 8, pp. 832–844, 1998.

[34] A. Criminisi, J. Shotton, and E. Konukoglu, "Decision forests: A unified framework for classification, regression, density estimation, manifold learning and semi-supervised learning," *Foundations and Trends® in Computer Graphics and Vision*, vol. 7, no. 2–3, pp. 81–227, 2012.

[35] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*. Springer, 2013, vol. 6.

[36] Y. Zhang, J. Xuan, B. G. de los Reyes, R. Clarke, and H. W. Ressom, "Reverse engineering module networks by pso-rnn hybrid modeling," *BMC genomics*, vol. 10, no. 1, p. S15, 2009.

[37] S. S. Haykin, S. S. Haykin, S. S. Haykin, and S. S. Haykin, *Neural networks and learning machines*. Pearson Upper Saddle River, NJ, USA:, 2009, vol. 3.

[38] J. L. Elman, "Finding structure in time," *Cognitive science*, vol. 14, no. 2, pp. 179–211, 1990.

[39] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence.* U Michigan Press, 1975.

[40] R. Poli, W. B. Langdon, N. F. McPhee, and J. R. Koza, *A field guide to genetic programming*. Lulu. com, 2008.

[41] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies–a comprehensive introduction," *Natural computing*, vol. 1, no. 1, pp. 3–52, 2002.

[42] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization*, vol. 11, no. 4, pp. 341–359, 1997.

[43] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 4, Nov 1995, pp. 1942–1948 vol.4.

[44] T. Rogalsky, S. Kocabiyik, and R. Derksen, "Differential evolution in aerodynamic optimization," *Canadian Aeronautics and Space Journal*, vol. 46, no. 4, pp. 183–190, 2000.

[45] R. Storn, "On the usage of differential evolution for function optimization," in *Fuzzy Information Processing Society, 1996. NAFIPS., 1996 Biennial Conference of the North American*. IEEE, 1996, pp. 519–523.

[46] A. P. Engelbrecht, *Computational intelligence: an introduction*. John Wiley & Sons, 2007.

[47] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 398–417, 2009.

[48] Z. Zhu, J. Zhou, Z. Ji, and Y.-H. Shi, "Dna sequence compression using adaptive particle swarm optimization-based memetic algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 5, pp. 643–658, 2011.

[49] K.-B. Lee and J.-H. Kim, "Multiobjective particle swarm optimization with preference-based sort and its application to path following footstep optimization for humanoid robots," *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 6, pp. 755–766, 2013.

[50] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360)*, May 1998, pp. 69–73.

[51] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE transactions on evolutionary computation*, vol. 10, no. 3, pp. 281–295, 2006.

[52] F. Spitz and E. E. Furlong, "Transcription factors: from enhancer binding to developmental control," *Nature Reviews Genetics*, vol. 13, no. 9, pp. 613–626, 2012.

[53] M. Levine and E. H. Davidson, "Gene regulatory networks for development," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, no. 14, pp. 4936–4942, 2005.

[54] H. D. Kim and E. K. O'Shea, "A quantitative model of transcription factor–activated gene expression," *Nature structural & molecular biology*, vol. 15, no. 11, pp. 1192–1198, 2008.

[55] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown, "Quantitative monitoring of gene expression patterns with a complementary dna microarray," *Science*, vol. 270, no. 5235, p. 467, 1995.

[56] T. R. Hughes, M. J. Marton, A. R. Jones, C. J. Roberts, R. Stoughton, C. D. Armour, H. A. Bennett, E. Coffey, H. Dai, Y. D. He *et al.*, "Functional discovery via a compendium of expression profiles," *Cell*, vol. 102, no. 1, pp. 109–126, 2000.

[57] T. I. Lee, N. J. Rinaldi, F. Robert, D. T. Odom, Z. Bar-Joseph, G. K. Gerber, N. M. Hannett, C. T. Harbison, C. M. Thompson, I. Simon *et al.*, "Transcriptional regulatory networks in saccharomyces cerevisiae," *science*, vol. 298, no. 5594, pp. 799–804, 2002.

[58] A. Blais and B. D. Dynlacht, "Constructing transcriptional regulatory networks," *Genes & development*, vol. 19, no. 13, pp. 1499–1511, 2005.

[59] D. S. Johnson, A. Mortazavi, R. M. Myers, and B. Wold, "Genome-wide mapping of in vivo protein-dna interactions," *Science*, vol. 316, no. 5830, pp. 1497–1502, 2007.

[60] P. J. Park, "Chip–seq: advantages and challenges of a maturing technology," *Nature Reviews Genetics*, vol. 10, no. 10, pp. 669–680, 2009.

[61] C. T. Workman, H. C. Mak, S. McCuine, J.-B. Tagne, M. Agarwal, O. Ozier, T. J. Begley, L. D. Samson, and T. Ideker, "A systems approach to mapping dna damage response pathways," *Science*, vol. 312, no. 5776, pp. 1054–1059, 2006.

[62] L. D. Bogarad, M. I. Arnone, C. Chang, and E. H. Davidson, "Interference with gene regulation in living sea urchin embryos: transcription factor knock out (tko), a genetically controlled vector for blockade of specific transcription factors," *Proceedings of the National Academy of Sciences*, vol. 95, no. 25, pp. 14 827–14 832, 1998.

[63] Z. Hu, P. J. Killion, and V. R. Iyer, "Genetic reconstruction of a functional transcriptional regulatory network," *Nature genetics*, vol. 39, no. 5, pp. 683–687, 2007.

[64] A. J. Hartemink, "Reverse engineering gene regulatory networks," *Nature biotechnology*, vol. 23, no. 5, pp. 554–555, 2005.

[65] M. Kaern, W. J. Blake, and J. J. Collins, "The engineering of gene regulatory networks," *Annual review of biomedical engineering*, vol. 5, no. 1, pp. 179–206, 2003.

[66] T. Tian, "Bayesian computation methods for inferring regulatory network models using biomedical data," in *Translational Biomedical Informatics*. Springer, 2016, pp. 289–307.

[67] R. J. Prill, D. Marbach, J. Saez-Rodriguez, P. K. Sorger, L. G. Alexopoulos, X. Xue, N. D. Clarke, G. Altan-Bonnet, and G. Stolovitzky, "Towards a rigorous assessment of systems biology models: the dream3 challenges," *PloS one*, vol. 5, no. 2, p. e9202, 2010.

[68] A. Greenfield, A. Madar, H. Ostrer, and R. Bonneau, "Dream4: Combining genetic and dynamic information to identify biological networks and dynamical models," *PloS one*, vol. 5, no. 10, p. e13397, 2010.

[69] D. Marbach, J. C. Costello, R. Küffner, N. M. Vega, R. J. Prill, D. M. Camacho, K. R. Allison, M. Kellis, J. J. Collins, G. Stolovitzky *et al.*, "Wisdom of crowds for robust gene network inference," *Nature methods*, vol. 9, no. 8, pp. 796–804, 2012.

[70] B. Godsey, "Improved inference of gene regulatory networks through integrated bayesian clustering and dynamic modeling of time-course expression data," *PloS one*, vol. 8, no. 7, p. e68358, 2013.

[71] K. Lu, R. Gordon, and T. Cao, "Reverse engineering the mechanical and molecular pathways in stem cell morphogenesis," *Journal of tissue engineering and regenerative medicine*, vol. 9, no. 3, pp. 169–173, 2015.

[72] C. Nicholson, L. Goodwin, and C. Clark, "Variable neighborhood search for reverse engineering of gene regulatory networks," *Journal of Biomedical Informatics*, 2016.

[73] J. Guinney, T. Wang, T. D. Laajala, K. K. Winner, J. C. Bare, E. C. Neto, S. A. Khan, G. Peddinti, A. Airola, T. Pahikkala *et al.*, "Prediction of overall survival for patients with metastatic castration-resistant prostate cancer: development of a prognostic model through a crowdsourced challenge with open clinical trial data," *The Lancet Oncology*, 2016.

[74] J. Tegner, M. S. Yeung, J. Hasty, and J. J. Collins, "Reverse engineering gene networks: integrating genetic perturbations with dynamical modeling," *Proceedings of the National Academy of Sciences*, vol. 100, no. 10, pp. 5944–5949, 2003.

[75] P. Dhaeseleer, S. Liang, and R. Somogyi, "Genetic network inference: from co-expression clustering to reverse engineering," *Bioinformatics*, vol. 16, no. 8, pp. 707–726, 2000.

[76] Y. Wang, T. Joshi, X.-S. Zhang, D. Xu, and L. Chen, "Inferring gene regulatory networks from multiple microarray datasets," *Bioinformatics*, vol. 22, no. 19, pp. 2413–2420, 2006.

[77] M. S. Yeung, J. Tegnér, and J. J. Collins, "Reverse engineering gene networks using singular value decomposition and robust regression," *Proceedings of the National Academy of Sciences*, vol. 99, no. 9, pp. 6163–6168, 2002.

[78] T. S. Gardner and J. J. Faith, "Reverse-engineering transcription control networks," *Physics of life reviews*, vol. 2, no. 1, pp. 65–88, 2005.

[79] J. Tegner, M. S. Yeung, J. Hasty, and J. J. Collins, "Reverse engineering gene networks: integrating genetic perturbations with dynamical modeling," *Proceedings of the National Academy of Sciences*, vol. 100, no. 10, pp. 5944–5949, 2003.

[80] Y. Wang, T. Joshi, X.-S. Zhang, D. Xu, and L. Chen, "Inferring gene regulatory networks from multiple microarray datasets," *Bioinformatics*, vol. 22, no. 19, pp. 2413–2420, 2006.

[81] B. P. Fairfax, S. Makino, J. Radhakrishnan, K. Plant, S. Leslie, A. Dilthey, P. Ellis, C. Langford, F. O. Vannberg, and J. C. Knight, "Genetics of gene expression in primary immune cells identifies cell type-specific master regulators and roles of hla alleles," *Nature genetics*, vol. 44, no. 5, pp. 502–510, 2012.

[82] Y. Pilpel, P. Sudarsanam, and G. M. Church, "Identifying regulatory networks by combinatorial analysis of promoter elements," *Nature genetics*, vol. 29, no. 2, pp. 153–159, 2001.

[83] K. C. Kao, Y.-L. Yang, R. Boscolo, C. Sabatti, V. Roychowdhury, and J. C. Liao, "Transcriptome-based determination of multiple transcription regulator activities in escherichia coli by using network component analysis," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 101, no. 2, pp. 641–646, 2004.

[84] F. Emmert-Streib, G. Glazko, R. De Matos Simoes *et al.*, "Statistical inference and reverse engineering of gene regulatory networks from observational expression data," *Frontiers in genetics*, vol. 3, p. 8, 2012.

[85] A. V. Werhli, M. Grzegorczyk, and D. Husmeier, "Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical gaussian models and bayesian networks," *Bioinformatics*, vol. 22, no. 20, pp. 2523–2531, 2006.

[86] W.-P. Lee and W.-S. Tzou, "Computational methods for discovering gene networks from expression data," *Briefings in bioinformatics*, vol. 10, no. 4, pp. 408–423, 2009.

[87] K. Basso, A. A. Margolin, G. Stolovitzky, U. Klein, R. Dalla-Favera, and A. Califano, "Reverse engineering of regulatory networks in human b cells," *Nature genetics*, vol. 37, no. 4, pp. 382–390, 2005.

[88] G. Stolovitzky, D. Monroe, and A. Califano, "Dialogue on reverse-engineering assessment and methods," *Annals of the New York Academy of Sciences*, vol. 1115, no. 1, pp. 1–22, 2007.

[89] T. Schaffter, D. Marbach, and D. Floreano, "Genenetweaver: in silico benchmark generation and performance profiling of network inference methods," *Bioinformatics*, vol. 27, no. 16, pp. 2263–2270, 2011.

[90] D. Marbach, R. J. Prill, T. Schaffter, C. Mattiussi, D. Floreano, and G. Stolovitzky, "Revealing strengths and weaknesses of methods for gene network inference," *Proceedings of the national academy of sciences*, vol. 107, no. 14, pp. 6286–6291, 2010.

[91] Z.-P. Liu and L. Chen, "Proteome-wide prediction of protein-protein interactions from high-throughput data," *Protein & cell*, vol. 3, no. 7, pp. 508–520, 2012.

[92] X. Zhang, X.-M. Zhao, K. He, L. Lu, Y. Cao, J. Liu, J.-K. Hao, Z.-P. Liu, and L. Chen, "Inferring gene regulatory networks from gene expression data by path consistency algorithm based on conditional mutual information," *Bioinformatics*, vol. 28, no. 1, pp. 98–104, 2012.

[93] M. Kaern, W. J. Blake, and J. J. Collins, "The engineering of gene regulatory networks," *Annual review of biomedical engineering*, vol. 5, no. 1, pp. 179–206, 2003.

[94] H. De Jong, "Modeling and simulation of genetic regulatory systems: a literature review," *Journal of computational biology*, vol. 9, no. 1, pp. 67–103, 2002.

[95] S. Wu, Z.-P. Liu, X. Qiu, and H. Wu, "Modeling genome-wide dynamic regulatory network in mouse lungs with influenza infection using high-dimensional ordinary differential equations," *PloS one*, vol. 9, no. 5, p. e95276, 2014.

[96] T. Lu, H. Liang, H. Li, and H. Wu, "High-dimensional odes coupled with mixed-effects modeling techniques for dynamic gene regulatory network identification," *Journal of the American Statistical Association*, 2012.

[97] M. Bansal, V. Belcastro, A. Ambesi-Impiombato, and D. Di Bernardo, "How to infer gene networks from expression profiles," *Molecular systems biology*, vol. 3, no. 1, p. 78, 2007.

[98] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.

[99] J. Fan and R. Li, "Variable selection via nonconcave penalized likelihood and its oracle properties," *Journal of the American statistical Association*, vol. 96, no. 456, pp. 1348–1360, 2001.

[100] F. Emmert-Streib, M. Dehmer, and B. Haibe-Kains, "Untangling statistical and biological models to understand network inference: the need for a genomics network ontology," *Frontiers in genetics*, vol. 5, p. 299, 2014.

[101] D. Marbach, J. C. Costello, R. Küffner, N. M. Vega, R. J. Prill, D. M. Camacho, K. R. Allison, M. Kellis, J. J. Collins, G. Stolovitzky *et al.*, "Wisdom of crowds for robust gene network inference," *Nature methods*, vol. 9, no. 8, pp. 796–804, 2012.

[102] Z.-P. Liu, W. Zhang, K. Horimoto, and L. Chen, "Gaussian graphical model for identifying significantly responsive regulatory networks from time course high-throughput data," *IET systems biology*, vol. 7, no. 5, pp. 143–152, 2013.

[103] H. Salgado, M. Peralta-Gil, S. Gama-Castro, A. Santos-Zavaleta, L. Muñiz-Rascado, J. S. García-Sotelo, V. Weiss, H. Solano-Lira, I. Martínez-Flores,

A. Medina-Rivera *et al.*, "Regulondb v8. 0: omics data sets, evolutionary conservation, regulatory phrases, cross-validated gold standards and more," *Nucleic acids research*, vol. 41, no. D1, pp. D203–D213, 2013.

[104] E. P. Consortium *et al.*, "An integrated encyclopedia of dna elements in the human genome," *Nature*, vol. 489, no. 7414, pp. 57–74, 2012.

[105] B. Ren, F. Robert, J. J. Wyrick, O. Aparicio, E. G. Jennings, I. Simon, J. Zeitlinger, J. Schreiber, N. Hannett, E. Kanin *et al.*, "Genome-wide location and function of dna binding proteins," *Science*, vol. 290, no. 5500, pp. 2306–2309, 2000.

[106] S. Mukherjee and T. P. Speed, "Network inference using informative priors," *Proceedings of the National Academy of Sciences*, vol. 105, no. 38, pp. 14 313–14 318, 2008.

[107] A. Bernard, A. J. Hartemink *et al.*, "Informative structure priors: joint learning of dynamic regulatory networks from multiple types of data." in *Pacific symposium on biocomputing*, vol. 10, 2005, pp. 459–470.

[108] A. V. Werhli and D. Husmeier, "Reconstructing gene regulatory networks with bayesian networks by combining expression data with multiple sources of prior knowledge," *Statistical applications in genetics and molecular biology*, vol. 6, no. 1, 2007.

[109] S. Imoto, T. Higuchi, T. Goto, K. Tashiro, S. Kuhara, and S. Miyano, "Combining microarrays and biological knowledge for estimating gene networks via bayesian networks," *Journal of bioinformatics and computational biology*, vol. 2, no. 01, pp. 77–98, 2004.

[110] Z.-P. Liu, X.-S. Zhang, and L. Chen, "Inferring gene regulatory networks from expression data with prior knowledge by linear programming," in *2010 International Conference on Machine Learning and Cybernetics*, vol. 6. IEEE, 2010, pp. 3067–3072.

[111] E. Charniak, "Bayesian networks without tears." *AI magazine*, vol. 12, no. 4, p. 50, 1991.

[112] A. Orun and H. Seker, "Development of a computer game-based framework for cognitive behaviour identification by using bayesian inference methods," *Computers in Human Behavior*, vol. 28, no. 4, pp. 1332–1341, 2012.

[113] A. Orun and N. Aydin, "Variable optimisation of medical image data by the learning bayesian network reasoning," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*. IEEE, 2010, pp. 4554–4557.

[114] C. E. Kahn, L. M. Roberts, K. A. Shaffer, and P. Haddawy, "Construction of a bayesian network for mammographic diagnosis of breast cancer," *Computers in biology and medicine*, vol. 27, no. 1, pp. 19–29, 1997.

[115] Z. Aydin, Y. Altunbasak, and H. Erdogan, "Bayesian models and algorithms for protein β-sheet prediction," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 8, no. 2, pp. 395–409, 2011.

[116] S. J. Russell, P. Norvig, J. F. Canny, J. M. Malik, and D. D. Edwards, *Artificial intelligence: a modern approach*. Prentice hall Upper Saddle River, 2003, vol. 2.

[117] N. Friedman, M. Linial, I. Nachman, and D. Pe'er, "Using bayesian networks to analyze expression data," *Journal of computational biology*, vol. 7, no. 3-4, pp. 601–620, 2000.

[118] V. A. Smith, E. D. Jarvis, and A. J. Hartemink, "Evaluating functional network inference using simulations of complex biological systems," *Bioinformatics*, vol. 18, no. suppl 1, pp. S216–S224, 2002.

[119] D. Heckerman, D. Geiger, and D. M. Chickering, "Learning bayesian networks: The combination of knowledge and statistical data," *Machine learning*, vol. 20, no. 3, pp. 197–243, 1995.

[120] M. Bansal and D. di Bernardo, "Inference of gene networks from temporal gene expression profiles," *IET Syst Biol*, vol. 1, no. 5, pp. 306–312, 2007.

[121] G. Arroyo-Figueroa and L. E. Sucar, "Temporal bayesian network of events for diagnosis and prediction in dynamic domains," *Applied Intelligence*, vol. 23, no. 2, pp. 77–86, 2005.

[122] U. Nodelman, C. R. Shelton, and D. Koller, "Continuous time bayesian networks," in *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2002, pp. 378–387.

[123] J. Yu, V. A. Smith, P. P. Wang, A. J. Hartemink, and E. D. Jarvis, "Advances to bayesian network inference for generating causal networks from observational biological data," *Bioinformatics*, vol. 20, no. 18, pp. 3594–3603, 2004.

[124] K. Murphy, S. Mian *et al.*, "Modelling gene expression data using dynamic bayesian networks," Technical report, Computer Science Division, University of California, Berkeley, CA, Tech. Rep., 1999.

[125] B.-E. Perrin, L. Ralaivola, A. Mazurie, S. Bottani, J. Mallet, and F. dAlche Buc, "Gene networks inference using dynamic bayesian networks," *Bioinformatics*, vol. 19, no. suppl 2, pp. ii138–ii148, 2003.

[126] S. Kim, S. Imoto, and S. Miyano, "Dynamic bayesian network and nonparametric regression for nonlinear modeling of gene networks from time series gene expression data," *Biosystems*, vol. 75, no. 1, pp. 57–65, 2004.

[127] M. Zou and S. D. Conzen, "A new dynamic bayesian network (dbn) approach for identifying gene regulatory networks from time course microarray data," *Bioinformatics*, vol. 21, no. 1, pp. 71–79, 2005.

[128] M. J. Beal, F. Falciani, Z. Ghahramani, C. Rangel, and D. L. Wild, "A bayesian approach to reconstructing genetic regulatory networks with hidden factors," *Bioinformatics*, vol. 21, no. 3, pp. 349–356, 2005.

[129] D. Husmeier, "Sensitivity and specificity of inferring genetic regulatory interactions from microarray experiments with dynamic bayesian networks," *Bioinformatics*, vol. 19, no. 17, pp. 2271–2282, 2003.

[130] S. Lebre, "Stochastic process analysis for genomics and dynamic bayesian networks inference." Ph.D. dissertation, Université d'Evry-Val d'Essonne, 2007.

[131] S. Lebre, original version 1.0 by Sophie Lebre, and contribution of Julien Chiquet to version 2.0, *G1DBN: A package performing Dynamic Bayesian Network inference.*, 2013, r package version 3.1.1. [Online]. Available: http://CRAN.R-project.org/package=G1DBN

[132] K. P. Murphy, "Dynamic bayesian networks: representation, inference and learning," Ph.D. dissertation, University of California, Berkeley, 2002.

[133] K. B. Korb and A. E. Nicholson, *Bayesian artificial intelligence*. CRC press, 2011.

[134] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[135] Z. Ghahramani, "Learning dynamic bayesian networks," in *Adaptive processing of sequences and data structures*. Springer, 1998, pp. 168–197.

[136] L. Rabiner and B. Juang, "An introduction to hidden markov models," *ieee assp magazine*, vol. 3, no. 1, pp. 4–16, 1986.

[137] M. Aymen, A. Abdelaziz, S. Halim, and H. Maaref, "Hidden markov models for automatic speech recognition," in *Communications, Computing and Control Applications (CCCA), 2011 International Conference on*. IEEE, 2011, pp. 1–6.

[138] J. Kim, S.-K. Lee, and B. Lee, "Classifying the speech response of the brain using gaussian hidden markov model (hmm) with independent component analysis (ica)," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE, 2013, pp. 4291–4294.

[139] G. Saon and J.-T. Chien, "Bayesian sensing hidden markov models for speech recognition," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2011, pp. 5056–5059.

[140] J.-C. Chen and J.-T. Chien, "Bayesian large margin hidden markov models for speech recognition," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2009, pp. 3765–3768.

[141] G. Megali, S. Sinigaglia, O. Tonet, and P. Dario, "Modelling and evaluation of surgical performance using hidden markov models," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 10, pp. 1911–1919, 2006.

[142] K. P. Murphy, *Machine learning: a probabilistic perspective*. MIT press, 2012.

[143] J. Hulst, "Modeling physiological processes with dynamic bayesian networks," Ph.D. dissertation, 2006.

[144] G. Welch and G. Bishop, "An introduction to the kalman filter. university of north carolina, department of computer science," TR 95-041, Tech. Rep., 1995.

[145] R. T. Sikora and K. Chauhan, "Estimating sequential bias in online reviews: A kalman filtering approach," *Knowledge-Based Systems*, vol. 27, pp. 314–321, 2012.

[146] E. Manla, A. Nasiri, C. H. Rentel, and M. Hughes, "Modeling of zinc bromide energy storage for vehicular applications," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 2, pp. 624–632, 2010.

[147] L. Tamas, M. Popa, G. Lazea, I. Szoke, and A. Majdik, "Lidar and vision based people detection and tracking," *Journal of Control Engineering and Applied Informatics*, vol. 12, no. 2, pp. 30–35, 2010.

[148] M. Manz, F. von Hundelshausen, and H.-J. Wuensche, "A hybrid estimation approach for autonomous dirt road following using multiple clothoid segments," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 2410–2415.

[149] P. Weckesser and R. Dillmann, "Modeling unknown environments with a mobile robot," *Robotics and Autonomous Systems*, vol. 23, no. 4, pp. 293–300, 1998.

[150] A. Calabrese and L. Paninski, "Kalman filter mixture model for spike sorting of non-stationary data," *Journal of neuroscience methods*, vol. 196, no. 1, pp. 159–169, 2011.

[151] B. C. Becker, R. A. MacLachlan, and C. N. Riviere, "State estimation and feedforward tremor suppression for a handheld micromanipulator with a kalman filter," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2011, pp. 5160–5165.

[152] K. Reif, S. Günther, E. Yaz, and R. Unbehauen, "Stochastic stability of the discrete-time extended kalman filter," *IEEE Transactions on Automatic control*, vol. 44, no. 4, pp. 714–728, 1999.

[153] G. Rigatos and S. Tzafestas, "Extended kalman filtering for fuzzy modelling and multi-sensor fusion," *Mathematical and computer modelling of dynamical systems*, vol. 13, no. 3, pp. 251–266, 2007.

[154] J. Ruess, A. Milias-Argeitis, S. Summers, and J. Lygeros, "Moment estimation for chemically reacting systems by extended kalman filtering," *The Journal of chemical physics*, vol. 135, no. 16, p. 165102, 2011.

[155] G. G. Rigatos, "Extended kalman and particle filtering for sensor fusion in motion control of mobile robots," *Mathematics and computers in simulation*, vol. 81, no. 3, pp. 590–607, 2010.

[156] T. M. Mitchell, *The discipline of machine learning*. Carnegie Mellon University, School of Computer Science, Machine Learning Department, 2006, vol. 9.

[157] S. Theodoridis and K. Koutroumbas, *Pattern Recognition, Fourth Edition*, 4th ed. Academic Press, 2008.

[158] A. Singh, N. Thakur, and A. Sharma, "A review of supervised machine learning algorithms," in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, March 2016, pp. 1310–1315.

[159] H. Pirim, B. Eksioglu, A. D. Perkins, and C. Yuceer, "Clustering of high throughput gene expression data," *COMPUTERS & OPERATIONS RESEARCH*, vol. 39, no. 12, pp. 3046–3061, DEC 2012.

[160] C.-T. Li, W.-S. Lai, C.-M. Liu, and Y.-F. Hsu, "Inferring reward prediction errors in patients with schizophrenia: a dynamic reward task for reinforcement learning," *Frontiers in psychology*, vol. 5, p. 1282, 2014.

[161] F. Zhu, Q. Liu, X. Zhang, and B. Shen, "Protein-protein interaction network constructing based on text mining and reinforcement learning with application to prostate cancer," in *Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on*. IEEE, 2014, pp. 46–51.

[162] A. Kara, M. Vickers, M. Swain, D. E. Whitworth, and N. Fernandez-Fuentes, "Metapred2cs: a sequence-based meta-predictor for protein–protein interactions of prokaryotic two-component system proteins," *Bioinformatics*, vol. 32, no. 21, pp. 3339–3341, 2016.

[163] G. L. Owens, K. Gajjar, J. Trevisan, S. W. Fogarty, S. E. Taylor, D. Gama-Rose, P. L. Martin-Hirsch, F. L. Martin *et al.*, "Vibrational biospectroscopy coupled with multivariate analysis extracts potentially diagnostic features in blood plasma/serum of ovarian cancer patients," *Journal of biophotonics*, vol. 7, no. 3-4, pp. 200–209, 2014.

[164] Q.-X. Wang, E.-D. Chen, Y.-F. Cai, Q. Li, Y.-X. Jin, W.-X. Jin, Y.-H. Wang, Z.-C. Zheng, L. Xue, O.-C. Wang *et al.*, "A panel of four genes accurately differentiates benign from malignant thyroid nodules," *Journal of Experimental & Clinical Cancer Research*, vol. 35, no. 1, p. 169, 2016.

[165] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial intelligence*, vol. 97, no. 1-2, pp. 273–324, 1997.

[166] S. P. Fodor, "Massively parallel genomics," *Science*, vol. 277, no. 5324, p. 393, 1997.

[167] G. E. Box, W. G. Hunter, J. S. Hunter *et al.*, "Statistics for experimenters," 1978.

[168] D. R. Cox and N. Reid, *The theory of the design of experiments*. CRC Press, 2000.

[169] T. Alexandrov, J. Decker, B. Mertens, A. M. Deelder, R. A. Tollenaar, P. Maass, and H. Thiele, "Biomarker discovery in maldi-tof serum protein profiles using discrete wavelet transformation," *Bioinformatics*, vol. 25, no. 5, pp. 643–649, 2009.

[170] D. Marbach, T. Schaffter, C. Mattiussi, and D. Floreano, "Generating realistic in silico gene networks for performance assessment of reverse engineering methods," *Journal of computational biology*, vol. 16, no. 2, pp. 229–239, 2009.

[171] GeneNetWeaver. (2010) Dream network inference challenge. [Online]. Available: http://gnw.sourceforge.net/dreamchallenge.html

[172] M. Radman, "Sos repair hypothesis: phenomenology of an inducible dna repair which is accompanied by mutagenesis," in *Molecular mechanisms for repair of DNA*. Springer, 1975, pp. 355–367.

[173] B. Michel, "After 30 years of study, the bacterial sos response still surprises us," *PLoS Biol*, vol. 3, no. 7, p. e255, 2005.

[174] X. Guo, Y. Zhang, W. Hu, H. Tan, and X. Wang, "Inferring nonlinear gene regulatory networks from gene expression data based on distance correlation," *PloS one*, vol. 9, no. 2, p. e87446, 2014.

[175] M. D. Adams, S. E. Celniker, R. A. Holt, C. A. Evans, J. D. Gocayne, P. G. Amanatides, S. E. Scherer, P. W. Li, R. A. Hoskins, R. F. Galle *et al.*, "The genome sequence of drosophila melanogaster," *Science*, vol. 287, no. 5461, pp. 2185–2195, 2000.

[176] J. Yang, Y. Zhu, H. Guo, X. Wang, R. Gao, L. Zhang, Y. Zhao, and X. Zhang, "Identifying serum biomarkers for ovarian cancer by screening with surface-enhanced laser desorption/ionization mass spectrometry and the artificial neural network," *International Journal of Gynecological Cancer*, vol. 23, no. 4, pp. 667–672, 2013.

[177] Z. Wei, Z. Xuan, and C. Junjie, "Study on classification rules of hypertension based on decision tree," in *Software Engineering and Service Science (ICSESS), 2013 4th IEEE International Conference on*. IEEE, 2013, pp. 93–96.

[178] S. Das, P. K. Ghosh, and S. Kar, "Hypertension diagnosis: a comparative study using fuzzy expert system and neuro fuzzy system," in *Fuzzy Systems (FUZZ), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1–7.

[179] T. Helleputte, *LiblineaR: Linear Predictive Models Based on the LIBLINEAR C/C++ Library*, 2015, r package version 1.94-2.

[180] N. Meinshausen and P. Bühlmann, "High-dimensional graphs and variable selection with the lasso," *The annals of statistics*, pp. 1436–1462, 2006.

[181] S. C. for Metabolomics, "Metlin: Metabolite and tandem ms database," Online, Jan. 2014. [Online]. Available: https://metlin.scripps.edu/index.php

[182] C. for Disease Control and P. (CDCP), "High blood pressure facts," Online, Apr. 2014. [Online]. Available: http://www.cdc.gov/bloodpressure/facts.htm

[183] B. H. Foundation, "High blood pressure," Online, Apr. 2014. [Online]. Available: https://www.bhf.org.uk/heart-health/risk-factors/high-blood-pressure

[184] U. N. L. o. M. National Center for Biotechnology Information, "Gene expression omnibus," Apr. 2014. [Online]. Available: https://www.ncbi.nlm.nih.gov/gene

[185] D. o. H. The National Health Service (NHS) UK, *Bowel Cancer*, 2016. [Online]. Available: http://www.nhs.uk/conditions/Cancer-of-the-colon-rectum-or-bowel/Pages/Introduction.aspx

[186] T. A. C. Soceity, *Key statistics for colorectal cancer*, 2016. [Online]. Available: http://www.cancer.org/cancer/colonandrectumcancer/detailedguide/colorectal-cancer-key-statistics

[187] S. Rathore, M. Hussain, A. Ali, and A. Khan, "A recent survey on colon cancer detection techniques," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 10, no. 3, pp. 545–563, 2013.

[188] Y. Hong, H. Wei, and L. Zeng-li, "Research for the colon cancer based on the emd and ls-svm," in *Intelligent Computation Technology and Automation (ICICTA), 2011 International Conference on*, vol. 1. IEEE, 2011, pp. 888–891.

[189] S. Rathore, M. A. Iftikhar, and M. Hussain, "A novel approach for automatic gene selection and classification of gene based colon cancer datasets," in *Emerging Technologies (ICET), 2014 International Conference on*. IEEE, 2014, pp. 42–47.

[190] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," *bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[191] A. Akutekwe and H. Seker, "Particle swarm optimization-based bio-network discovery method for the diagnosis of colorectal cancer," in *2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, Nov 2014, pp. 8–13.

[192] T. Jirapech-Umpai and S. Aitken, "Feature selection and classification for microarray data analysis: Evolutionary methods for identifying predictive genes," *BMC bioinformatics*, vol. 6, no. 1, p. 1, 2005.

[193] C. Ooi and P. Tan, "Genetic algorithms applied to multi-class prediction for the analysis of gene expression data," *Bioinformatics*, vol. 19, no. 1, pp. 37–44, 2003.

[194] K.-B. Duan, J. C. Rajapakse, H. Wang, and F. Azuaje, "Multiple svm-rfe for gene selection in cancer classification with expression data," *IEEE transactions on nanobioscience*, vol. 4, no. 3, pp. 228–234, 2005.

[195] X. Zhou and D. P. Tuck, "Msvm-rfe: extensions of svm-rfe for multiclass gene selection on dna microarray data," *Bioinformatics*, vol. 23, no. 9, pp. 1106–1114, 2007.

[196] C. X. Ling, J. Huang, and H. Zhang, "Auc: a better measure than accuracy in comparing learning algorithms," in *Conference of the Canadian Society for Computational Studies of Intelligence*. Springer, 2003, pp. 329–341.

[197] N.-J. Fan, R. Kang, X.-Y. Ge, M. Li, Y. Liu, H.-M. Chen, and C.-F. Gao, "Identification alpha-2-hs-glycoprotein precursor and tubulin beta chain as serology diagnosis biomarker of colorectal cancer," *Diagnostic pathology*, vol. 9, no. 1, p. 1, 2014.

[198] A. Fernández-Grijalva, A. Aguilar-Lemarroy, L. Jave-Suarez, A. Gutiérrez-Ortega, P. Godinez-Melgoza, S. Herrera-Rodríguez, I. Mariscal-Ramírez, M. Martínez-Velázquez, M. Gawinowicz, M. Martínez-Silva *et al.*, "Alpha 2hs-glycoprotein, a tumor-associated antigen (taa) detected in mexican patients with early-stage breast cancer," *Journal of proteomics*, vol. 112, pp. 301–312, 2015.

[199] H.-J. Son, J. W. Park, H. J. Chang, D. Y. Kim, B. C. Kim, S. Y. Kim, S. C. Park, H. S. Choi, and J. H. Oh, "Preoperative plasma hyperfibrinogenemia is predictive of poor prognosis in patients with nonmetastatic colon cancer," *Annals of surgical oncology*, vol. 20, no. 9, pp. 2908–2913, 2013.

[200] J. Wrangle, E. O. Machida, L. Danilova, A. Hulbert, N. Franco, W. Zhang, S. C. Glöckner, M. Tessema, L. Van Neste, H. Easwaran *et al.*, "Functional identification of cancer-specific methylation of cdo1, hoxa9, and tac1 for the diagnosis of lung cancer," *Clinical Cancer Research*, vol. 20, no. 7, pp. 1856–1864, 2014.

[201] W. Cao, H. Liu, X. Liu, L. J. LI JG, M. LIU, and Z. NIU, "Relaxin enhances in-vitro invasiveness of breast cancer cell lines by upregulation of s100a4/mmps signaling," *Eur Rev Med Pharmacol Sci*, vol. 17, no. 5, pp. 609–617, 2013.

[202] K. Pazaitou-Panayiotou, C. Chemonidou, A. Poupi, M. Koureta, A. Kaprara, M. Lambropoulou, T. C. Constantinidis, G. Galaktidou, M. Koffa, A. Kiziridou

et al., "Gonadotropin-releasing hormone neuropeptides and receptor in human breast cancer: Correlation to poor prognosis parameters," *Peptides*, vol. 42, pp. 15–24, 2013.

[203] S. L. Poon, C. Klausen, G. L. Hammond, and P. C. Leung, "37-kda laminin receptor precursor mediates gnrh-ii–induced mmp-2 expression and invasiveness in ovarian cancer cells," *Molecular Endocrinology*, vol. 25, no. 2, pp. 327–338, 2010.

[204] S. Ling Poon, M.-T. Lau, G. L. Hammond, and P. C. Leung, "Gonadotropin-releasing hormone-ii increases membrane type i metalloproteinase production via β-catenin signaling in ovarian cancer cells," *Endocrinology*, vol. 152, no. 3, pp. 764–772, 2011.

[205] R. Stovold, A. Stevens, D. Ray, P. Sommers, C. Dive, F. Blackhall, and A. White, "Pro-opiomelanocortin is a novel biomarker for small cell lung cancer," 2010.

[206] G. G. Malouf, S. Job, V. Paradis, M. Fabre, L. Brugières, P. Saintigny, L. Vescovo, J. Belghiti, S. Branchereau, S. Faivre *et al.*, "Transcriptional profiling of pure fibrolamellar hepatocellular carcinoma reveals an endocrine signature," *Hepatology*, vol. 59, no. 6, pp. 2228–2237, 2014.

[207] H. Yamagishi, D. H. Fitzgerald, T. Sein, T. J. Walsh, and B. C. O'Connell, "Saliva affects the antifungal activity of exogenously added histatin 3 towards candida albicans," *FEMS microbiology letters*, vol. 244, no. 1, pp. 207–212, 2005.

[208] U. Consortium *et al.*, "Uniprot: a hub for protein information," *Nucleic acids research*, p. gku989, 2015.

[209] A. Muendlein, M. Hubalek, S. Geller-Rhomberg, K. Gasser, T. Winder, H. Drexel, T. Decker, E. Mueller-Holzner, M. Chamson, C. Marth *et al.*, "Significant survival impact of macc1 polymorphisms in her2 positive breast cancer patients," *European Journal of Cancer*, vol. 50, no. 12, pp. 2134–2141, 2014.

[210] D. Ji, Z.-T. Lu, Y.-Q. Li, Z.-Y. Liang, P.-F. Zhang, C. Li, J.-L. Zhang, X. Zheng, and Y.-M. Yao, "Macc1 expression correlates with pfkfb2 and survival in hepatocellular carcinoma." *Asian Pacific journal of cancer prevention: APJCP*, vol. 15, no. 2, pp. 999–1003, 2013.

[211] M.-C. Paquin, C. Leblanc, E. Lemieux, B. Bian, and N. Rivard, "Functional impact of colorectal cancer-associated mutations in the transcription factor e2f4," *International journal of oncology*, vol. 43, no. 6, pp. 2015–2022, 2013.

[212] R. Colucci, C. Blandizzi, M. Tanini, C. Vassalle, M. C. Breschi, and M. D. Tacca, "Gastrin promotes human colon cancer cell growth via cck-2 receptor-mediated cyclooxygenase-2 induction and prostaglandin e2 production," *British journal of pharmacology*, vol. 144, no. 3, pp. 338–348, 2005.

[213] T. Hasegawa, R. Yamaguchi, M. Nagasaki, S. Miyano, and S. Imoto, "Inference of gene regulatory networks incorporating multi-source biological knowledge via a state space model with l 1 regularization," *PloS one*, vol. 9, no. 8, p. e105942, 2014.

[214] N. D. Lawrence, G. Sanguinetti, and M. Rattray, "Modelling transcriptional regulation using gaussian processes," in *Advances in Neural Information Processing Systems*, 2006, pp. 785–792.

[215] S. Y. Kim, S. Imoto, and S. Miyano, "Inferring gene networks from time series microarray data using dynamic bayesian networks," *Briefings in bioinformatics*, vol. 4, no. 3, pp. 228–235, 2003.

[216] K. Kojima, S. Imoto, R. Yamaguchi, A. Fujita, M. Yamauchi, N. Gotoh, and S. Miyano, "Identifying regulational alterations in gene regulatory networks by state space representation of vector autoregressive models and variational annealing," *BMC genomics*, vol. 13, no. 1, p. 1, 2012.

[217] S. Imoto, T. Goto, S. Miyano *et al.*, "Estimation of genetic networks and functional structures between genes by using bayesian networks and nonparametric regression," in *Pacific symposium on Biocomputing*, vol. 7, 2001, pp. 175–186.

[218] A. A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. D. Favera, and A. Califano, "Aracne: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context," *BMC bioinformatics*, vol. 7, no. Suppl 1, p. S7, 2006.

[219] X. Yang, J. E. Dent, and C. Nardini, "An s-system parameter estimation method (spem) for biological networks," *Journal of Computational Biology*, vol. 19, no. 2, pp. 175–187, 2012.

[220] K. Nakamura, R. Yoshida, M. Nagasaki, S. Miyano, and T. Higuchi, "Parameter estimation of in silico biological pathways with particle filtering towards a petascale computing." in *Pacific Symposium on Biocomputing*, vol. 14, 2009, pp. 227–238.

[221] C. A. Penfold and D. L. Wild, "How to infer gene networks from expression profiles, revisited," *Interface focus*, vol. 1, no. 6, pp. 857–870, 2011.

[222] R. Köker, "Design and performance of an intelligent predictive controller for a six-degree-of-freedom robot using the elman network," *Information Sciences*, vol. 176, no. 12, pp. 1781–1799, 2006.

[223] C. Ni and X. Yan, "Elman neural networks with sensitivity pruning for modeling fed-batch fermentation processes," *Journal of chemical engineering of Japan*, vol. 48, no. 3, pp. 230–237, 2015.

[224] C. Bergmeir and J. M. Benítez, "Neural networks in R using the stuttgart neural network simulator: RSNNS," *Journal of Statistical Software*, vol. 46, no. 7, pp. 1–26, 2012. [Online]. Available: http://www.jstatsoft.org/v46/i07/

[225] J. J. Faith, B. Hayete, J. T. Thaden, I. Mogno, J. Wierzbowski, G. Cottarel, S. Kasif, J. J. Collins, and T. S. Gardner, "Large-scale mapping and validation of escherichia coli transcriptional regulation from a compendium of expression profiles," *PLoS biol*, vol. 5, no. 1, p. e8, 2007.

[226] Z. Lv, W. Liu, D. Li, L. Liu, J. Wei, J. Zhang, Y. Ge, Z. Wang, H. Chen, C. Zhou *et al.*, "Association of functional fen1 genetic variants and haplotypes and breast cancer risk," *Gene*, vol. 538, no. 1, pp. 42–45, 2014.

[227] L. Sun, X. Sun, S. Xie, H. Yu, and D. Zhong, "Significant decrease of adp release rate underlies the potent activity of dimethylenastron to inhibit mitotic kinesin eg5 and cancer cell proliferation," *Biochemical and biophysical research communications*, vol. 447, no. 3, pp. 465–470, 2014.

[228] M. Harada, Y. Kotake, T. Ohhata, K. Kitagawa, H. Niida, S. Matsuura, K. Funai, H. Sugimura, T. Suda, and M. Kitagawa, "Yb-1 promotes transcription of cyclin d1 in human non-small-cell lung cancers," *Genes to Cells*, vol. 19, no. 6, pp. 504–516, 2014.

[229] J. Chiquet, A. Smith, G. Grasseau, C. Matias, and C. Ambroise, "Simone: Statistical inference for modular networks," *Bioinformatics*, vol. 25, no. 3, pp. 417–418, 2009.

[230] J. Schäfer, R. Opgen-Rhein, and K. Strimmer, "Reverse engineering genetic networks using the genenet package," *J Am Stat Assoc*, vol. 96, pp. 1151–1160, 2001.